AB-IR-03-96

EULER ANGLES AND QUATERNIONS IN SIX DEGREE OF FREEDOM SIMULATIONS OF PROJECTILES

NO AD#

by

Michael J. Amoruso

March 1996

BEST AVAILABLE COPY

Approved for public release;
distribution unlimited.

REVIEWED BY: ______________________
Sung-Kwong Chung
Aerospace Engineer, Aeroballistics Branch, MATD, AED

APPROVED BY: ______________________
Dr. William Ebihara
Chief, Materials & Aeroballistics Technology Division, AED

AEROBALLISTICS BRANCH
MATERIALS & AEROBALLISTICS TECHNOLOGY DIVISION
ARMAMENT ENGINEERING DIRECTORATE
ARMAMENT RESEARCH, DEVELOPMENT AND ENGINEERING CENTER
U.S. ARMY ARMAMENT, MUNITIONS AND CHEMICAL COMMAND
PICATINNY ARSENAL, NEW JERSEY 07806-5000

CONTENTS

FIGURES

TABLES

PREFACE

The author has been involved in the simulation of guided projectiles for many years. Different investigators adopt different coordinate frames, such as body-fixed, plane-fixed and aeroballistic (zero P). Some use the Euler angle representation to deal with rotations and some use quaternions. Sources which explain the significance and advantages and disadvantages of these various approaches are not readily available. It is difficult to find derivations and there is a lack of advice on incorporating these methods into computer simulations. The author became especially frustrated when he attempted to collect the equations to convert an existing six degree of freedom (6 DOF) simulation from the Euler angle to the quaternion representation. Several sources for the needed equations were found but no two agreed exactly. Since little in the way of derivations were provided, it was not trivial to verify the equations or reconcile the discrepancies. This document resulted from the author's attempt to make some sense of this confusion.

The first chapter contains an overview of the problem. In the second chapter, a brief review of the bare minimum of matrix algebra is provided to remind the reader of some of the important properties of orthogonal transformations. The third chapter develops the Euler angle formalism with an introduction to the difference between body-fixed, plane-fixed and aeroballistic coordinates. The quaternion algebra is developed in chapter 4. This is an extensive subject. Only enough of the formalism was developed to provide understanding of quaternions and introduce the tools needed for this document. In chapter 5 the rigid body equations of motion are developed for the three coordinate frames discussed, in both the Euler angle and quaternion representations. The discussion of the distinction between body-fixed, plane-fixed and aeroballistic coordinates is distributed throughout chapters 3 to 5. In chapter 6, the integration of the equations of motion is discussed. Discussions of the treatment of Coriolis and centripetal corrections in a flat earth model, gravity for a non-flat earth, and time varying mass and moment of inertia have been included in this report. These topics will be treated in a future report. The appendix provides a summary of the algorithms needed for implementing these results in a 6 DOF simulation.

The author wishes to thank Dr. Richard Haddad of Polytechnic University of New York, Messrs Romel Campbell and John Grau of ARDEC, Dover, NJ, and Mr. Thomas Harkins of ARL, Aberdeen, MD, for valuable discussions and suggestions. He wishes to thank Mr Sung Chung for checking the mathematical derivations.

1 INTRODUCTION

When developing simulations of aircraft, missiles or gun-launched projectiles, investigators require a coordinate frame in which to follow the motion. Newton's laws require an inertial (unaccelerated) frame. The earth is a convenient reference frame but is not inertial since the earth rotates. The earth may nontheless be used, with Coriolis and centripetal accelerations included to account for the earth's rotation.

However, the projectile is both translating and rotating. Thus it is convenient to express the equations of motion of the projectile, missile or aircraft in coordinates that move along with it in some way. The obvious choice is body-fixed coordinates. These coordinates are attached to the projectile or aircraft and roll, pitch and yaw with it. The reader familiar with gimbals or gyroscopes will recognize that these Euler angles of roll, pitch and yaw are equivalent to gimbal angles. In the case of a guided projectile, the seeker, rate sensor, accelerometers, and control mechanisms whether aerodynamic or reaction control all operate in and are easiest to describe in body-fixed coordinates.

Sometimes non rolling coordinates are desirable. It is difficult to interpret results of a simulation when the point of view is rolling, as they are with body-fixed coordinates. In addition, spin-stabilized gun fired projectiles rotate at hundreds of revolutions per second. Computer run times for such projectiles using body-fixed coordinates become intolerably long. This difficulty arises because the integration time step must become extremely small in order to keep the angle of roll small during the integration time step. If this is not done, gravity is smeared over the angular motion that occurs during the integration time step because of the high roll rate, giving incorrect results.

Some type of non-rolling coordinate system is used to deal with this problem. One solution is to set the x component of the coordinate frame angular velocity to zero. Another is to set the Euler roll angle to zero. These two approaches are not identical, as we shall see in subsequent chapters. We shall see that the difference arises from the fact that the components of the angular velocity form an orthogonal set whereas the three Euler angles do not have a mutually orthogonal set of rotation axes.

Choosing the roll Euler angle to be zero eliminates the horizontal component of gravity in a flat earth model entirely since we shall see that the y-axis is constrained to move in a horizontal plane. This makes the numerical integration insensitive to the roll rate. However, it is still sensitive to the pitch and yaw rates. This approach is typically selected when modeling an unguided stage of a spin stabilized projectile. This type of frame is called plane-fixed.

Choosing the x component of the coordinate frame angular velocity to be zero yields aeroballistic coordinates. This choice does not completely eliminate the y component of gravity but sensitivity to the effects of roll is greatly reduced. Its chief value is the simplification of the equations of motion. Coupling terms

involving the x component of the frame angular velocity disappear from the equations of motion. If further simplifications are made based on symmetry and linearity of the aerodynamics, it is possible to obtain closed form solutions to the equations of motion[1].

With any of these frames, it is necessary to regenerate the frame as the projectile moves. Thus the frame itself has equations of motion. We shall see that the rotation matrix that transforms the vectors from the moving frame to the inertial (earth) frame can be expressed in terms of either three Euler angles or four quaternions. The equations of motion for the Euler angles and for the quaternions are derived so that they may be integrated to obtain the new frame and update the projectile equations of motion. Only two angles are required to describe the rotation of a rigid body so not all the Euler angles or the quaternions are linearly independent. Constraints such as normalization conditions therefore exist and will be derived.

The advantage of Euler angles over quaternions is their intuitiveness. Roll, pitch and yaw are a natural way for a pilot to describe or visualize the angular motion of an aircraft. The Euler angles are the natural variable for describing a seeker or spinning gyroscope gimbal. However the Euler angle algebra is somewhat messy and unsymmetrical, so errors are not always evident. Furthermore, the sine and cosine of the three Euler angles must be repeated computed, providing a computational burden that does not exist with quaternions. Thus, although quaternions are not intuitive in the sense that Euler angles are, their simplicity and symmetric form make derivations much simpler, are less prone to mask errors and are computationally more efficient. No trigonometric functions or transcendental functions need to be evaluated. The most complicated quaternion arithmetic requires the square of a quaternion or the product of two quaternions. For this reason quaternion algebra is desirable in digital autopilots for guided projectiles because it alleviates the computational burden. Furthermore, Euler angles are susceptible to singularities that can be avoided by using the quaternion formalism.

The details of the Euler angle and quaternion formalism required to develop the 6 DOF equations of motion for a rigid body in the three coordinate frames discussed above will be developed in subsequent chapters.

---

[1] Vaughn, Harold R., "A detailed Development of the Tricyclic Theory," Sandia Laboratories, SC-M-67-2933, Albuquerque, NM, 1968.

2 REVIEW OF MATRIX ALGEBRA FOR ORTHOGONAL TRANSFORMATIONS

This chapter contains a brief review of the matrix algebra required in this document[1,2]. An n by m matrix **A** is an array of elements a_{ij}, i = 1 to n, j = 1 to m, of i rows and j columns, which obeys the following laws of addition and multiplication,

$$\mathbf{C} = \mathbf{A} + \mathbf{B} \quad \rightarrow \quad c_{ij} = a_{ij} + b_{ij} \tag{2.1}$$

$$\mathbf{C} = \mathbf{A}\,\mathbf{B} \quad \rightarrow \quad c_{ij} = \sum_{j} a_{ik}\, b_{kj} \tag{2.2}$$

For these operations to be meaningful, certain matching restrictions exist on the number of rows and columns. For addition, **A**, **B** and **C** must all have the same number of rows and the same number of columns. For multiplication, the number of columns of **A** must match the number of rows of **B**. The product **C** has the same number of rows as **A** and the same number of columns as **B**. Such matrices are said to be conformable .If the number of rows and columns are equal, the matrix is square. A vector can be represented by an n by 1 column matrix.

The usual algebraic laws hold except that multiplication is not generally commutative and the multiplicative inverse does not always exist (see below). When we refer to the inverse of a square matrix, we generally mean the multiplicative inverse. The inverse of a square matrix **A** is denoted by $\mathbf{A}^{-1}$ and is defined by

$$\mathbf{A}\,\mathbf{A}^{-1} = \mathbf{A}^{-1}\,\mathbf{A} = \mathbf{1} \tag{2.3}$$

where **1** is the unit matrix (i.e., 1 along the diagonal and zero elsewhere). This can also be written in terms of the elements of the matrix

$$\sum_{j} a_{ij}\, a_{jk}^{-1} = \sum_{j} a_{ij}^{-1}\, a_{jk} = \delta_{ik} \tag{2.4}$$

---

1 Wylie, C. R., Jr, "Advanced Engineering Mathematics," McGraw-Hill, New York, 1956.

2 Margenau, Henry and George Moseley Murphy, "The Mathematics of Physics and Chemistry," Van Nostrand, New York, 1956.

where i,j and k = 1, 2, 3 and δ_{ik} is the Kronecker delta function which is unity when the subscripts are equal and zero otherwise. In general, the inverse of a matrix does not always exist. Generally, a necessary and sufficient condition for a matrix to have an inverse is that it must be square and the determinant of the matrix must be non zero. Such a matrix is called non singular. Even when a matrix is non singular, finding the inverse is generally tedious. This will not concern us here since we will be dealing only with orthogonal matrices which are always square and non-singular. We shall also see that the inverse can be found quite trivially. From this point onwards, we adopt the summation convention which states that a sum is implied over any index that is repeated twice. Thus

$$a_{ij}\, b_{jk} = \sum_j a_{ij}\, b_{jk} \tag{2.5}$$

We define an orthogonal matrix as one that preserves the length of a vector upon which it operates. The matrix operators than rotate rigid-body vectors must preserve length since rotation does not stretch or compress a rigid body. Thus a rotational transformation must be a subset of orthogonal transformations. What properties can we derive from this definition? Consider the matrix **A** operating on the column matrix or vector **v**.

$$\mathbf{v}' = \mathbf{A}\ \mathbf{v} \tag{2.6}$$

This may also be written in terms of the elements

$$v' = a_{ij}\, v_j \tag{2.7}$$

If the length of **v** is preserved by this transformation **A**, then

$$\mathbf{v}' \cdot \mathbf{v}' = \mathbf{v} \cdot \mathbf{v} \tag{2.8}$$

where the dot product is defined by

$$\mathbf{v} \cdot \mathbf{v} = \mathbf{v}_1^2 + \mathbf{v}_2^2 + \mathbf{v}_3^2 \tag{2.9}$$

Thus, if **A** is an orthogonal, length-preserving transformation

$$v_i\, v_i = v'_i\, v'_i = a_{ij} v_j\, a_{ik} v_k = a_{ij} a_{ik} v_j v_k \tag{2.10}$$

This is possible only if

$$a_{ij} a_{ik} = \delta_{jk} \tag{2.11}$$

But this is the definition of the inverse of **A**.

$$a_{ji}^{-1}\, a_{ik} = \delta_{jk} \tag{2.12}$$

Thus, for an orthogonal matrix, $a_{ji}^{-1} = a_{ij}$. The inverse of an orthogonal matrix

is obtained by interchanging the subscripts or the rows and columns of the matrix. This is equivalent to reflecting the matrix about the diagonal. This operation is called the transpose and will be denoted by a superscript T. Thus

$$\mathbf{A}^{-1} = \mathbf{A}^{T} \tag{2.13}$$

defines an orthogonal matrix [1].

We define the trace or spur of a matrix. The trace is simply the sum of all the elements along the diagonal. Thus

$$\mathbf{Tr}\ [\ \mathbf{A}\ ] = \mathbf{Tr}\ [\ \mathbf{A}^{T}\ ] = \sum_{i} a_{ii} \tag{2.14}$$

The trace is obviously invariant to the transpose operation since the diagonal elements are unchanged under transposition.

Finally, we show that the transpose of a product of two matrices is the product of the transposes of the individaul matrices, but in reverse order.

$$[\ \mathbf{A}\ \mathbf{B}\ ]^{T} = \mathbf{B}^{T}\ \mathbf{A}^{T} \tag{2.15}$$

The proof is as follows:

$$[\ \mathbf{A}\ \mathbf{B}\ ]_{ik}^{T} = [\ \mathbf{A}_{iq}\ \mathbf{B}_{qk}\ ]^{T} = \mathbf{A}_{kq}\,\mathbf{B}_{qi}$$

$$\mathbf{B}_{qi}\ \mathbf{A}_{kq} = \mathbf{B}_{iq}^{T}\ \mathbf{A}_{qk}^{T} = [\ \mathbf{B}^{T}\ \mathbf{A}^{T}\ ]_{ik} \tag{2.16}$$

---

[1] It can be shown that the determinant is invariant to the transpose operation. Therefore from (2.3) and (2.13) we can conclude that the square of the determinant of an orthogonal matrix is 1. Thus the determinant must be ± 1. The negative determinant is associated with reflections, which obviously also preserves length. The positive determinant is associated with rotations.

3 EULER ANGLES

3.1 ROTATING COORDINATE FRAMES

We use a right hand coordinate system with x positive forward, y positive to the right and z positive down, as indicated in Figure 1. Each Euler angle also obeys a right hand rule with respect to its axis. The roll φ occurs about the x axis, the pitch θ occurs about the y axis, and the yaw ψ occurs about the z axis. Thus the roll is positive clockwise looking forward from the rear, pitch is positive upward (even though z is positive down), and the yaw is positive looking forward. By convention, the rotational transformation from inertial to rotating axes consists of a yaw through angle ψ, followed by a pitch through angle θ, and finally a roll through angle φ. The order of these rotations is important since rotations do not commute. The component rotations are shown in Figure 1. The original inertial axes are indicated by x, y, and z. The primes indicate intermediate axes of subsequent rotations. The final rotating axes are indicated by x'', y''', and z'''. These component rotations may be expressed as

$$R_{\psi} = \begin{vmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{vmatrix} \tag{3.1}$$

$$R_{\theta} = \begin{vmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{vmatrix} \tag{3.2}$$

$$R_{\phi} = \begin{vmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{vmatrix} \tag{3.3}$$

---

1 Blakelock, John H., "Automatic Control of Aircraft and Missiles," John Wiley and Sons, New York, 1965.

2 Etkin, Bernard, "Dynamics of Flight - Stability and Control," John Wiley and Sons, New York, 1965.

3 Etkin, Bernard, "Dynamics of Atmospheric Flight," John Wiley and Sons, New York, 1972.

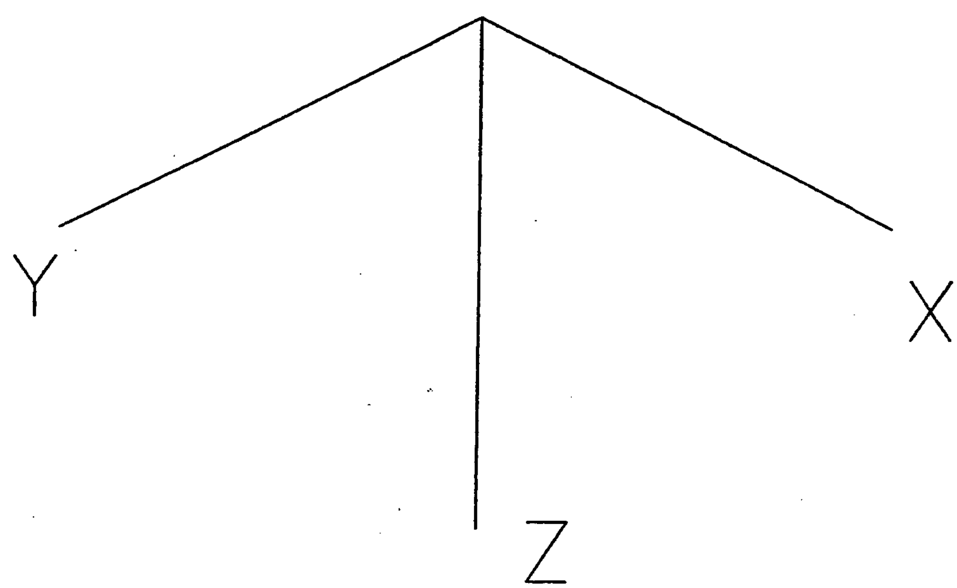
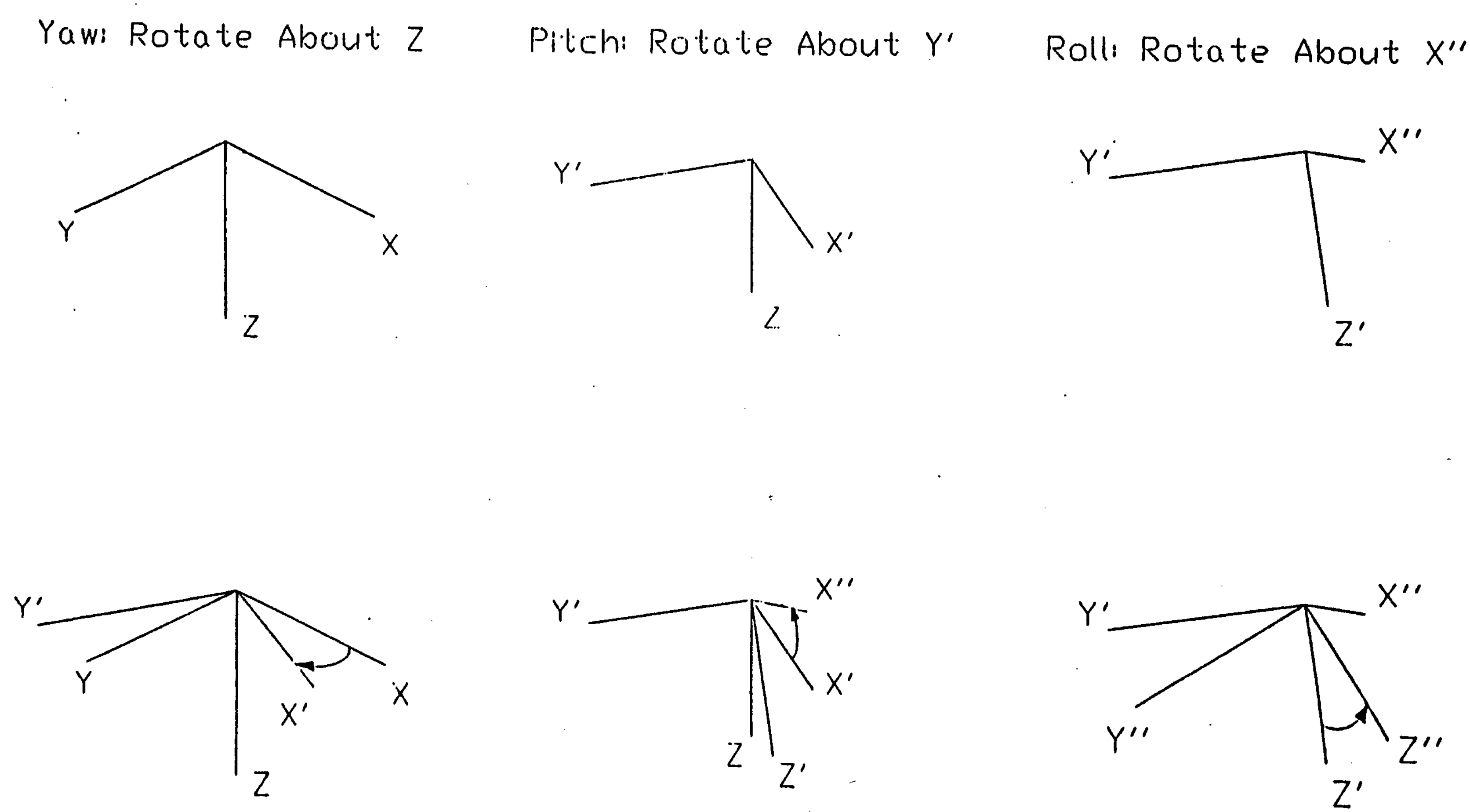


Figure 1. Coordinate System



Inertial frame to moving frame rotation. Yaw followed by pitch followed by roll.

The effect of yaw on the coordinate frame is shown on the left. The original frame position is at the top. The effect of yaw is shown at the bottom. Similarly for pitch and roll.

Figure 2. Component Euler Angle Rotations

$$0 \leq \phi < 2\pi \,, \quad \frac{-\pi}{2} \leq \theta \leq \frac{\pi}{2} \,, \quad -\pi < \psi \leq \pi \tag{3.4}$$

It is conventional to think of a rotation from the non-moving (e.g., inertial) coordinates to the moving body-fixed coordinates in terms of a yaw followed by a pitch and finally a roll. However, the more commonly used transformation in simulations will be the inverse rotation from body-fixed to non-moving coordinates. To find this transformation, we must reverse both the order of the component subrotations and reverse the algebraic sign of the roll, pitch and yaw angles. Thus the matrix rotation that transforms a vector from the rotating axes to the inertial axes is obtained by evaluating the matrix product

$$\mathbf{T} = R_{-\psi} R_{-\theta} R_{-\phi} =$$

$$\left| \begin{array}{ccc} \cos\theta \cos\psi & \begin{array}{l} -\cos\phi \sin\psi \\ +\sin\phi \sin\theta \cos\psi \end{array} & \begin{array}{l} \sin\phi \sin\psi \\ +\cos\phi \sin\theta \cos\psi \end{array} \\ \cos\theta \sin\psi & \begin{array}{l} \cos\phi \cos\psi \\ +\sin\phi \sin\theta \sin\psi \end{array} & \begin{array}{l} -\sin\phi \cos\psi \\ +\cos\phi \sin\theta \sin\psi \end{array} \\ -\sin\theta & \sin\phi \cos\theta & \cos\phi \cos\theta \end{array} \right| \tag{3.5}$$

Since this matrix is a rotation and therefore orthogonal, the inverse transformation from inertial to rotating coordinates is obtained by taking the transpose, i.e., interchanging rows and columns. Later on we shall distinguish between various types of moving coordinate systems: body-fixed, plane-fixed, and aeroballistic coordinates.

Another interesting set of expressions that is needed is the components of the angular velocity Ω of the coordinate axes in terms of the Euler angles and their derivatives. This will be useful in the development of the equations of motion. An uncritical guess might be that the components of Ω are nothing more than the derivatives of the three Euler angles. While it is true that Ω is the vector sum of the vectors associated with these Euler angle rates, these rate vectors are not mutually orthogonal and so can not be the components. However, they can be expressed in such a way that they can be resolved into orthogonal components in the moving frame. In general, the rotation associated with Ω can be considered as consisting of three successive non-orthogonal rotations with angular velocities. See Figure 2. In the following, a bar denotes a vector and a hat denotes a unit vector.

$$\bar{\Omega} = \bar{\Omega}_{\psi} + \bar{\Omega}_{\theta} + \bar{\Omega}_{\phi} \tag{3.6}$$

$$= \dot{\psi}\,\hat{z} + \dot{\theta}\,\hat{y}' + \dot{\phi}\,\hat{x}''$$

We want to resolve this in moving-fixed coordinates, where we can write

$$\bar{\Omega} = \Omega_{x''}\,\hat{x}'' + \Omega_{y'''}\,\hat{y}''' + \Omega_{z'''}\,\hat{z}''' \tag{3.7}$$

The vector quantities on the right of (3.6) are not an orthogonal set whereas those in (3.7) are orthogonal. The unit vectors $\hat{x}''$, $\hat{y}'''$ and $\hat{z}'''$ are orthogonal. These may be resolved as follows. The vector Ω_{ψ} resolved in the orthogonal coordinates in the moving frame is obtained by applying the partial Euler rotation:

$$\Omega_{\psi} = \begin{vmatrix} \left(\Omega_{\psi}\right)_{x''} \\ \left(\Omega_{\psi}\right)_{y'''} \\ \left(\Omega_{\psi}\right)_{z'''} \end{vmatrix} = R_{\phi}\,R_{\theta} \begin{vmatrix} 0 \\ 0 \\ \dot{\psi} \end{vmatrix}$$

$$= \begin{vmatrix} -\dot{\psi}\, sin(\theta) \\ \dot{\psi}\, sin(\phi)\, cos(\theta) \\ \dot{\psi}\, cos(\phi)\, cos(\theta) \end{vmatrix} \tag{3.8}$$

Likewise,

$$\Omega_{\theta} = \begin{vmatrix} \left(\Omega_{\theta}\right)_{x''} \\ \left(\Omega_{\theta}\right)_{y'''} \\ \left(\Omega_{\theta}\right)_{z'''} \end{vmatrix} = R_{\phi} \begin{vmatrix} 0 \\ \dot{\theta} \\ 0 \end{vmatrix} \tag{3.9}$$

$$= \begin{vmatrix} 0 \\ +\dot{\theta}\cos(\phi) \\ -\dot{\theta}\sin(\phi) \end{vmatrix}$$

No transformation is needed for the components of Ω_ϕ since it is parallel to the x''-axis. Thus

$$\left(\Omega_\phi\right)_{x''} = \dot{\phi} \tag{3.10}$$

$$\left(\Omega_\phi\right)_{y'''} = 0 \tag{3.11}$$

$$\left(\Omega_\phi\right)_{z'''} = 0 \tag{3.12}$$

We can add vectorially (3.8) to (3.10) to obtain an explicit representation of (3.6) resolved in the moving frame. Comparing this to (3.7) gives the result we seek, viz.

$$\Omega_{x''} = -\dot{\psi}\, sin(\theta) + \dot{\phi} \tag{3.13}$$

$$\Omega_{y'''} = \dot{\psi}\, sin(\phi)\, cos(\theta) + \dot{\theta}\, cos(\phi) \tag{3.14}$$

$$\Omega_{z'''} = \dot{\psi}\, cos(\phi)\, cos(\theta) - \dot{\theta}\, sin(\phi) \tag{3.15}$$

These can be inverted for the derivatives of the Euler angles by the usual algebraic techniques to get (dropping the primes)

$$\dot{\psi} = \frac{\left(\Omega_y\, sin(\phi) + \Omega_z\, cos(\phi)\right)}{cos(\theta)} \tag{3.16}$$

$$\dot{\phi} = \Omega_x + \left(\Omega_y \, sin(\phi) + \Omega_z \, cos(\phi)\right) tan(\theta) \tag{3.17}$$

$$\dot{\theta} = \Omega_y \, cos(\phi) - \Omega_z \, sin(\phi) \tag{3.18}$$

In a computer simulation, equations (3.16) through (3.18) are numerically integrated to update the three Euler angles. These are used in turn to update the rotation matrix (3.5). Note that the above expressions have a singularity at $\theta = \pi/2$. Euler angles can be chosen so that the singularity occurs elsewhere, but there is always a singularity. In simulations using Euler angles, care must be taken that the vicinity of the singularity is avoided.

By inspection, (3.16) through (3.18) can be put into matrix form.

$$\begin{vmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{vmatrix} = \begin{vmatrix} 1 & 0 & -\sin(\theta) \\ 0 & \cos(\phi) & \sin(\phi)\cos(\theta) \\ 0 & -\sin(\phi) & \cos(\phi)\cos(\theta) \end{vmatrix} \begin{vmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{vmatrix} \tag{3.19}$$

Likewise, (3.16) through (3.18) may be written from inspection as

$$\begin{vmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{vmatrix} = \begin{vmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi)/\cos(\theta) & \cos(\phi)/\cos(\theta) \end{vmatrix} \begin{vmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{vmatrix} \tag{3.20}$$

Expressions (3.19) and (3.20) are not symmetrical or elegant. Note that the matrices in (3.19) and (3.20) must be inverses of one another. This can be verified by taking the product of these two matrices and verifying that the unit identity matrix results.

The choice for the coordinate frame angular velocity components Ω_x, Ω_y, and Ω_z will depend on the application. Body-fixed coordinates are appropriate for simulating a guided projectile, rocket or aircraft. In a guided projectile the seeker, sensors and control mechanisms are fixed to the body. For this reason it is easiest and simplest to describe these subsystems in a coordinate frame fixed to the projectile body. The coordinate frame angular velocity components are then equal to the analogous components of the body angular velocity, which are conventionally denoted by P, Q and R. Thus in equations (3.13) through (3.20), we make the substitutions

$$\begin{aligned} \Omega_x &= P \\ \Omega_y &= Q \\ \Omega_z &= R \end{aligned} \tag{3.21}$$

This is the usual choice made for 6 DOF simulations of guided projectiles and missiles.

For unguided projectiles, a non-rolling coordinate frame is often preferred. Such a frame pitches and yaws with the projectile but does not roll with it. A non-rolling frame might be defined by letting the x component of the frame angular velocity, Ω_x, vanish or by letting the time derivative of the roll Euler angle, ϕ, vanish. From (3.13) we see that these two approaches are not identical. The coordinates obtained by the first approach are called aeroballistic coordinates and the latter plane-fixed coordinates. The advantage of the former is he simplification of the equations of motion since the coupling terms involving Ω_x drop out, as we shall see in Chapter 5. This approach is often taken with analytic or closed-form solutions of the equations of motion. Equation (3.21) would be modified by letting $\Omega_x = 0$.

The plane-fixed approach is often used for 6 DOF computer simulations of spin stabilized projectiles. Spin stabilized projectiles have typical spin rates of hundreds of revolutions per second. Using a body-fixed representation in a computer simulation of such a projectile requires an extremely small integration time step and, consequently, inordinately long computer run times. The time step must be small so that the projectile roll is not appreciable during the time step. Otherwise the effect of gravity is smeared across this angle. While aeroballistic coordinates will help, a more useful solution is to require $d\phi/dt = \dot{\phi} = 0$ for the coordinate frame. We shall see in Chapter 5 that the y component of gravity is rigorously eliminated in the fixed plane approach. This eliminates sensitivity of the integration to the projectile roll rate, though a similar sensitivity is still present for the much slower pitch and yaw rates. This approach can speed up simulation run time by orders of magnitude.

All three approaches will be discussed further when developing the equations of motion in Chapter 5. The plane-fixed coordinates are derived in the following

section from the physical view point of constraining one of the axes to move in a single plane.

3.2 PLANE-FIXED COORDINATES

Plane-fixed coordinates pitch and yaw with the body but do not roll with the body. Hence we define $\phi = 0$ and $d\phi/dt = 0$. More precisely, one axis is constrained to always remain in one plane, though it can rotate in that plane. For example, the z-axis could be constrained to the vertical plane (original x-z plane). This can be achieved in an inertial to moving (i.e., plane-fixed) frame transformation consisting in a pitch about the original y-axis (which keeps the z-axis in the original pitch plane, which is vertical) followed by a yaw about the new z-axis, which leaves the z-axis unchanged and therefore still in the vertical plane.

Alternatively, the y-axis can be constrained to the horizontal plane (original x-y plane). This is done by a yaw about the z-axis followed by a pitch about the y-axis. We can construct the rotation matrix as before, using (3.1) and (3.2). However, unlike before, the above recipe involves inverse transformations from the inertial to the moving frame rather than from moving frame to inertial, as was the case in the derivation of (3.5) for the body-fixed frame. Thus the above transformation is comprised of the inverses of the matrix operators previously used. Thus, recalling (2.15) and that the inverse of an orthogonal matrix is its transpose,

$$\mathbf{T}^{-1} = \mathbf{R}_\theta^{-1}\,\mathbf{R}_\psi^{-1} = \mathbf{R}_\theta^{T}\,\mathbf{R}_\psi^{T} = [\,\mathbf{R}_\psi\,\mathbf{R}_\theta\,]^{T} = \mathbf{T}^{T}$$

or

$$\mathbf{T} = \mathbf{R}_\psi\,\mathbf{R}_\theta =$$

$$\begin{vmatrix} \cos\theta\,\cos\psi & -\sin\psi & \sin\theta\,\cos\psi \\ \cos\theta\,\sin\psi & \cos\psi & \sin\theta\,\sin\psi \\ -\sin\theta & 0 & \cos\theta \end{vmatrix} \tag{3.22}$$

This is the plane-fixed analog of (3.5). It is not surprising that this is equivalent to making ϕ vanish in (3.5). Likewise (3.6) becomes

$$\overline{\Omega} = \overline{\Omega}_{\psi} + \overline{\Omega}_{\theta} = \dot{\psi}\,\hat{z} + \dot{\theta}\,\hat{y}' \tag{3.23}$$

Equation (3.8) becomes

$$\Omega_{\psi} = \begin{vmatrix} \left(\Omega_{\psi}\right)_{x''} \\ \left(\Omega_{\psi}\right)_{y'''} \\ 0 \end{vmatrix} = R_{\theta} \begin{vmatrix} 0 \\ 0 \\ \dot{\psi} \end{vmatrix} .$$

$$= \begin{vmatrix} -\dot{\psi}\, sin(\theta) \\ \dot{0} \\ +\dot{\psi}\, cos(\theta) \end{vmatrix} \tag{3.24}$$

Likewise, (3.9) becomes

$$\Omega_{\theta} = \begin{vmatrix} \left(\Omega_{\theta}\right)_{x''} \\ \left(\Omega_{\theta}\right)_{y'''} \\ 0 \end{vmatrix} = \begin{vmatrix} 0 \\ \dot{\theta} \\ 0 \end{vmatrix}$$

Equations (3.13) through (3.15) become

$$\Omega_{z'''} = \dot{\psi}\, cos(\theta) \tag{3.26}$$

$$\Omega_{y'''} = \dot{\theta} \tag{3.27}$$

$$\Omega_{x''} = -\dot{\psi}\, sin(\theta) = -\Omega_{z'''}\, \tan(\theta) \tag{3.28}$$

We have substituted (3.26) into (3.28). The primes on the subscripts of Ω in (3.26) through (3.28), as well as in (3.13) through (3.15) can be dropped since the axes referred to are orthogonal. Inverting, with some algebra yields the analogs of (3.16) through (3.18), viz.,

$$\dot{\psi} = \frac{\Omega_z}{cos(\theta)} = \frac{-\Omega_x}{\sin(\theta)} \tag{3.29}$$

$$\dot{\phi} = 0 \tag{3.30}$$

$$\dot{\theta} = \Omega_y \tag{3.31}$$

The analogs of (3.19) and (3.20) are

$$\begin{vmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{vmatrix} = \begin{vmatrix} 1 & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ 0 & 0 & \cos(\theta) \end{vmatrix} \begin{vmatrix} 0 \\ \dot{\theta} \\ \dot{\psi} \end{vmatrix} \tag{3.32}$$

and

$$\begin{vmatrix} 0 \\ \dot{\theta} \\ \dot{\psi} \end{vmatrix} = \begin{vmatrix} 1 & 0 & \tan(\theta) \\ 0 & 1 & 0 \\ 0 & 0 & 1/\cos(\theta) \end{vmatrix} \begin{vmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{vmatrix} \tag{3.33}$$

In summary, the representation of the rotation matrix T is unique no matter how it is derived, although some methods may not always work because of singularities. The rotation matrix T can be viewed as an operator which rotates a vector in a fixed coordinate system or, conversely, as a rotation of the coordinate system while the vector remains fixed. In the former point of view, the vector has the same length but its components are changed because its direction in space has changed due to the vector's rotation. In the latter point of view, the vector has the same length and direction in space but its components are different because of the rotation of the coordinate frame.

If we take the latter point of view, we can see that the rotation matrix is just the matrix of the direction cosines. Let $\hat{\imath}'_p$ denote the three mutually orthogonal unit basis vectors of the primed (rotated) coordinate system and $\hat{\imath}_q$ denote the three basis vectors of the unprimed coordinate frame. T_{pq} will be the elements of the transformation from the unprimed to the primed frame. Then

$$\hat{\imath}'_p = T_{pq}\, \hat{\imath}_q \tag{3.34}$$

Taking the dot product and making use of the mutual orthogonality of the basis vectors, we can write

$$\begin{aligned} \hat{\imath}'_p \cdot \hat{\imath}_l &= T_{pq}\, \hat{\imath}_q \cdot \hat{\imath}_l \\ &= T_{pq}\, \delta_{ql} = T_{pl} \end{aligned} \tag{3.35}$$

Thus the elements of the rotation matrix could be obtained by taking the dot products of the unit basis vectors of the unprimed coordinate frame with the basis vectors of the primed frame.

4 QUATERNIONS

We will develop just enough of the algebra of quaternions as is needed for understanding and writing six degree-of-freedom (6 DOF) simulations. Quaternions are a quadruplet of numbers (strictly speaking operators) that can be considered to be a generalization of complex numbers. Recall that the quantity

$$i = \sqrt{-1} \tag{4.1}$$

may be best thought of as a rotation operator. Thus i is a 90 degree counterclockwise rotation from the "real" to the "imaginary" axis. See Figure 3. The "square" of i is two successive 90 degree rotations. This is equivalent to a 180 degree rotation. This takes us to the negative real axis. The cube of i is a 270 degree counterclockwise rotation from the positive real to the negative imaginary axis. It is in this sense that $i^2 = -1$ and $i^3 = -i$. The 4th power of i is just a 360 degree rotation which gets us back to the real axis.

For quaternions we define three such quantities, corresponding to rotations about the x, y, and z axes respectively. As in the case of the conventional imaginary number i, the operators i, j, and k can be interpreted as 90 degree rotations about the x, y, and z axes; and the squares correspond to a 180 degree rotation about the appropriate direction, and so forth. These operators are sometimes call hyperimaginary numbers. Just as the conventional complex numbers can be used to provide machinery for treating rotations in a plane, we might expect that three "imaginary" operators i, j, and k might be used to treat rotations about three axes, i.e., in three dimensional space. Another useful property of quaternions is that it permits us to multiply and divide vectors. This will be seen to provide a much simpler mathematical treatment than matrices.

Recall that rotations are not numbers but operators and do not commute. Thus ij does not equal ji, and so forth. A little bit of thought and some experimentation with rotations will convince the reader that the following elementary relationships hold.

$$i^2 = j^2 = k^2 = -1 \tag{4.2}$$

$$i\,j = k = -j\,i \tag{4.3}$$

$$j\,k = i = -k\,j \tag{4.4}$$

$$k\,i = j = -i\,k \tag{4.5}$$

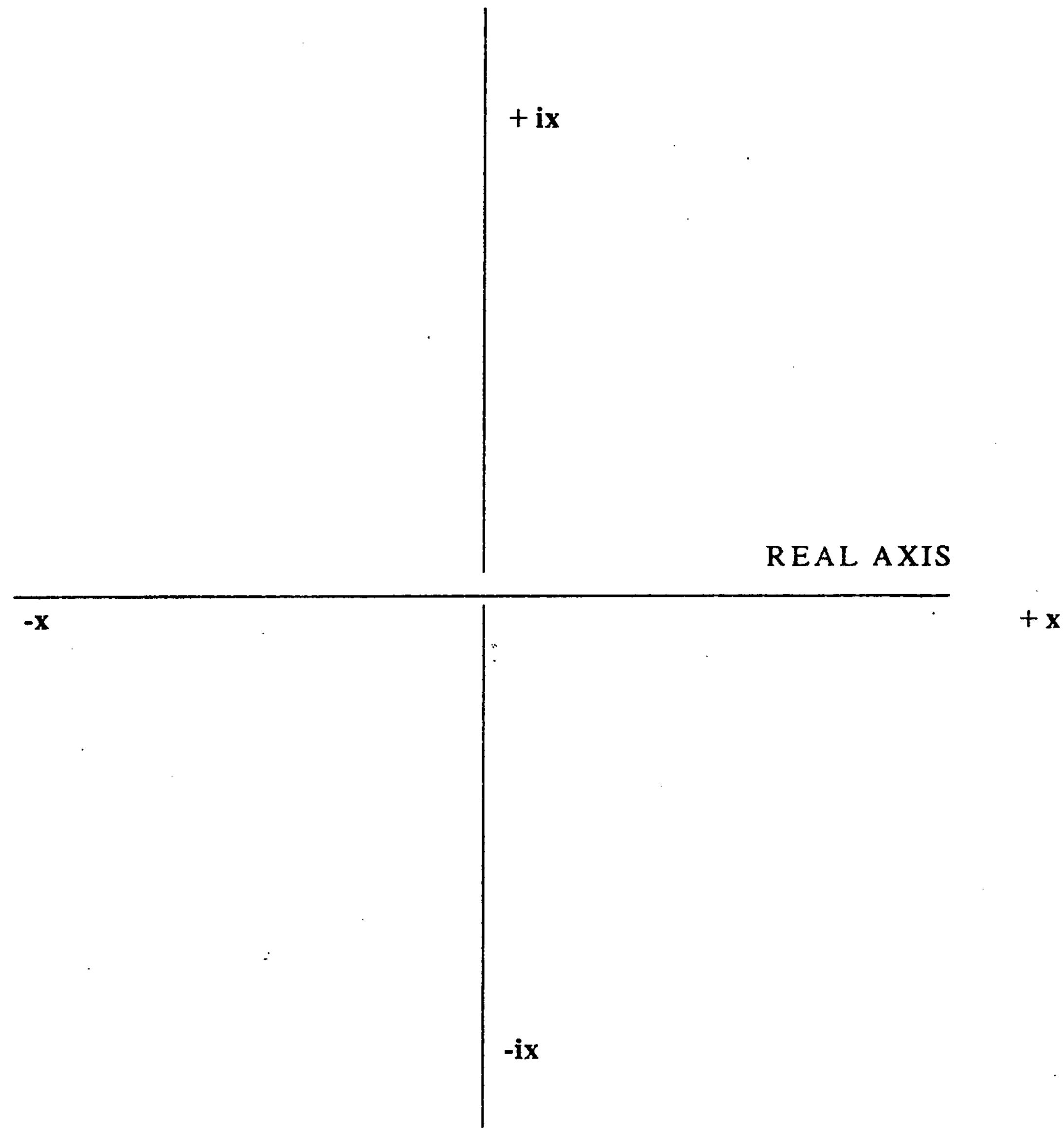


Figure 3. Imaginary Number i Interpreted as a Rotation Operator

We take a quaternion to be a quadruple $\mathbf{Q} = [\ Q_0 + iQ_1 + jQ_2 + kQ_3\ ]$. Some tedious algebra will verify that multiplying out two quaternions, taking all possible products and simplifying using (4.2) through (4.5), gives the following result.

$$\mathbf{W}\ \mathbf{Q} = [\ W_0 + iW_1 + jW_2 + kW_3\ ]\ [\ Q_0 + iQ_1 + jQ_2 + kQ_3\ ] \tag{4.6}$$

$$= [W_0Q_0 - W_1Q_1 - W_2Q_2 - W_3Q_3] + W_0[iQ_1 + jQ_2 + kQ_3] + Q_0[iW_1 + jW_2 + kW_3]$$

$$+ i[\ W_2Q_3 - W_3Q_2\ ] + j[\ W_3Q_1 - W_1Q_3\ ] + k[\ W_1Q_2 - W_2Q_1\ ]$$

Upon inspection of (4.6), we see that the first line after the last equality contains expressions that resemble a dot product and a cross product. This suggests an alternative formulation. We can treat i, j and k not as rotation operators or hyperimaginary numbers but as an orthogonal set of unit vectors and consider a quaternion formally to consist of a scalar part and a vector part. This lets us write the quaternion product expressed in (4.6) more compactly.

The quaternion product is equivalently defined as

$$\mathbf{W}\ \mathbf{Q} = [\ W_0Q_0 - \overline{W}\cdot\overline{Q}\ ,\ W_0\,\overline{Q} + Q_0\overline{W} + \overline{W}\times\overline{Q}\ ] \tag{4.7}$$

where

$$\mathbf{W} \equiv [\ W_0\ ,\ \overline{W}\ ] \text{ and } \mathbf{Q} \equiv [\ Q_0\ ,\ \overline{Q}\ ] \tag{4.8}$$

Furthermore, we can obtain still another alternate form since (4.6) can also be rearranged into

$$\begin{aligned} \mathbf{W}\ \mathbf{Q} = \quad & [\ W_0Q_0 - W_1Q_1 - W_2Q_2 - W_3Q_3\ ] \\ & + i[\ W_1Q_0 + W_0Q_1 - W_3Q_2 + W_2Q_3\ ] \\ & + j[\ W_2Q_0 + W_3Q_1 + W_0Q_2 - W_1Q_3\ ] \\ & + k[\ W_3Q_0 - W_2Q_1 + W_1Q_2 + W_0Q_3\ ] \end{aligned} \tag{4.9}$$

This may be organized into the matrix form as

$$\mathbf{W\,Q} \equiv \mathbf{A} =$$

$$\begin{vmatrix} A_0 \\ A_1 \\ A_2 \\ A_3 \end{vmatrix} = \begin{vmatrix} +W_0 & -W_1 & -W_2 & -W_3 \\ +W_1 & +W_0 & -W_3 & +W_2 \\ +W_2 & +W_3 & +W_0 & -W_1 \\ +W_3 & -W_2 & +W_1 & +W_0 \end{vmatrix} \begin{vmatrix} Q_0 \\ Q_1 \\ Q_2 \\ Q_3 \end{vmatrix} \tag{4.10}$$

$$= \begin{vmatrix} +Q_0 & -Q_1 & -Q_2 & -Q_3 \\ +Q_1 & +Q_0 & +Q_3 & -Q_2 \\ +Q_2 & -Q_3 & +Q_0 & +Q_1 \\ +Q_3 & +Q_2 & -Q_1 & +Q_0 \end{vmatrix} \begin{vmatrix} W_0 \\ W_1 \\ W_2 \\ W_3 \end{vmatrix}$$

Compare (4.7) and (4.10) to (4.6). They are equivalent. (4.7) and (4.10) are more compact than (4.6) but are arbitrary if used as a definition for quaternion multiplication, as some authors do. On the other hand, (4.6) flows logically and automatically from the properties (4.1) to (4.5), and gives insight into the connection with rotations. We will use the approach that is the most convenient in each case.

Some useful expressions follow. Define the conjugate

$$\mathbf{Q}^* \equiv [\, Q_0 \,, -\bar{Q} \,] \tag{4.11}$$

Define the norm or absolute value squared

$$|\,\mathbf{Q}\,|^2 \equiv \mathbf{Q}\,\mathbf{Q}^* = [\, Q_0^2 + \bar{Q}\cdot\bar{Q} \,, 0\,] = \mathbf{Q}^*\,\mathbf{Q} \tag{4.12}$$

Let us define an invese and verify it works.

$$\mathbf{Q}^{-1} \equiv [\, Q_0 \,, -\bar{Q} \,] \,/\, (\, Q_0^2 + \bar{Q}\cdot\bar{Q} \,)$$

$$= \mathbf{Q}^* \,/\, |\,\mathbf{Q}\,| \tag{4.13}$$

It follows that

$$Q\,Q^{-1} = [\,1\,,0\,] \tag{4.14}$$

If the norm vanishes, the quaternion is said to be singular and the inverse does not exist. It is easy to show that the norm of a product equals the product of the norms. The inverse of a product is the product of the inverses in reverse order. The conjugate of a product is the product of the conjugates in reverse order. Thus

$$|\,Q_1\,Q_2\,| = |\,Q_1\,|\,|\,Q_2\,| \tag{4.15}$$

$$[\,Q_1\,Q_2\,]^{-1} = Q_2^{\;-1}\,Q_1^{\;-1} \tag{4.16}$$

$$[\,Q_1\,Q_2\,]^{*} = Q_2^{\;*}\,Q_1^{\;*} \tag{4.17}$$

In general, quaternion arithmetic will be familiar except for non-commutativity of multiplication. Commutation breaks down for multiplication because of the cross product term in (4.7). Otherwise all the other usual laws are obeyed. Quaternion arithmetic is distributive and associative, but commutative only for addition. Identity elements exist for both addition and multiplication, viz. (0,0) and (1,0). Inverses also exist for addition for all quaternions. A multiplicative inverse exists for any non-zero quaternion. See (4.13). The rules for differentiation are the familiar ones, except care must be taken because quaternions do not commute. As an example, consider the derivative of the product of two quaternions.

$$[\,Q_1\,Q_2\,]' = Q_1'\,Q_2 + Q_1\,Q_2' \neq Q_1'\,Q_2 + Q_2'\,Q_1 \tag{4.18}$$

Since the norm of a product equals the product of the norms, it would seem plausible that a quaternion of unit norm could be useful to treat rotations since rotations must preserve the length of vectors. We shall see later that this conjecture is essentially correct. In anticipation of a quaternion formalism for rotation, some relations for unit quaternions will now be provided.

Consider the unit quaternion (sometimes called a versor or Euler quaternion)

$$\mathbf{e} = [\,e_0\,,\bar{e}\,] = [\,e_0\,,e_1\,e_2\,e_3\,] \tag{4.19}$$

where $|\,\mathbf{e}\,|^2 \equiv [\,1\,,0\,]$ See (4.12). Thus

$$e_0^2 + e_1^2 + e_2^2 + e_3^2 = 1 \tag{4.20}$$

From (4.15), the "length" or norm of a quaternion is preserved when multiplied by a unit quaternion.

$$|\mathbf{e}\,\mathbf{Q}| = |\mathbf{Q}| \tag{4.21}$$

Since the norm of a unit quaternion is unity, (4.13) tells us that the inverse of a unit quaternion is equal to its conjugate, i.e.

$$\mathbf{e}^{-1} = [\, e_0 \,,\, -\bar{e} \,] = [\, e_0 \,,\, -e_1 \, -e_2 \, -e_3 \,] \; = \mathbf{e}^{*} \tag{4.22}$$

Also

$$\dot{e} \; = \; \left[\dot{e}_0 \,,\, \dot{e}_1 \;\; \dot{e}_2 \;\; \dot{e}_3 \right] \tag{4.23}$$

By differentiating $e\ e^{-1} = 1$, we obtain

$$\dot{e}\ e^{-1} = -e\ \dot{e}^{-1} \tag{4.24}$$

Applying (4.16) to (4.24) gives

$$\begin{aligned} \dot{e}\ e^{-1} &= -\left[\dot{e}\ e^{-1} \right]^{-1} \\ &= \left[e\ \dot{e}^{-1} \right]^{-1} \end{aligned} \tag{4.25}$$

Since the inverse of a unit quaternion is equal to its conjugate, the inverses in (4.25) can be replaced by conjugation.

$$\begin{aligned} \dot{e}\ e^{*} &= -\left[\dot{e}\ e^{*} \right]^{*} \\ &= \left[e\ \dot{e}^{*} \right]^{*} \end{aligned} \tag{4.26}$$

Vectors can be treated as quaternions with a zero scalar part. Note that

$$\mathbf{V}^* = [\, 0 \,, \bar{\mathbf{v}} \,]^* = [\, 0 \,, -\bar{\mathbf{v}} \,] = -\, \mathbf{V} \tag{4.27}$$

Some authors refer to a quaternion that has a zero scalar part as a pure quaternion. Generally, the quaternion product of two pure quaternions (vectors) is not a pure quaternion (vector) since the scalar part of the product is usually not zero. Sometimes the product of two quaternions with non-zero scalar part yields a pure quaternion. For example, from (4.26)

$$\dot{e}\, e^* = -\left[\dot{e}\, e^*\right]^* \tag{4.28}$$

This is only possible if the above product is hyperimaginary, i.e., a pure quaternion or vector.

We now try to formulate a rotation operator in terms of quaternions operating on a vector (i.e., a quaternion with a zero scalar part). The simplest thing to try is multiplication of a vector from the right or left by a unit quaternion. A unit quaternion is chosen since rotations preserve the length or norm of a vector, and we can see from (4.21) that multiplication by a unit quaternion will not change length. First we will try multiplication from the left. We shall see that multiplication from the left only (or from the right only) is unsatisfactory.

We choose for the rotation operator the unit quaternion

$$\lambda = \left[\lambda_0 \,, \bar{\lambda}\right] \equiv \left[\cos(\beta) \,,\ \sin(\beta)\, \hat{\lambda}\right] \tag{4.29}$$

where the "hat" denotes a unit vector and

$$\lambda_0^2 + \bar{\lambda}\cdot\bar{\lambda} = 1 = \lambda_0^2 + \lambda_1^2 + \lambda_2^2 + \lambda_3^2 \tag{4.30}$$

For the vector we choose a quaternion representation with zero scalar part

$$\mathbf{q} = \left[0 \,, \bar{q}\right] \tag{4.31}$$

Then we try representing a rotation by

$$\mathbf{q}' = \lambda\, \mathbf{q} = \left[\lambda_0 \,, \bar{\lambda}\right] \left[0 \,, \bar{q}\right] = \tag{4.32}$$

$$\left[-\bar{\lambda}\cdot\bar{q} \ , \ \lambda_0\bar{q} + \bar{\lambda}\times\bar{q} \right]$$

For quaternion **q′** to be a "vector", the scalar part must vanish. But multiplying a pure vector by a quaternion produces another quaternion with a non-vector component. Thus, unless we make an orthogonality assumption that $\lambda\cdot\bar{q} = 0$ so the scalar part vanishes, the quaternion multiplication does too much. But such an assumption is too restrictive. Alternatively, we could try multiplying by λ from the right. This gives

$$\mathbf{q}' = \mathbf{q}\,\lambda = \left[0 \ , \ \bar{q}\right]\left[\lambda_0 \ , \ \bar{\lambda}\right] = \left[-\bar{q}\cdot\bar{\lambda} \ , \ \lambda_0\bar{q} + \bar{q}\times\bar{\lambda} \right] \tag{4.33}$$

Note that, from (4.22) we can write

$$\mathbf{q}' = \mathbf{q}\,\lambda^{-1} = \left[+\bar{q}\cdot\bar{\lambda} \ , \ -\lambda_0\bar{q} + \bar{\lambda}\times\bar{q} \right] \tag{4.34}$$

The scalar parts of (4.32) and (4.34) have opposite sign. This suggests we might try combining (4.32) and (4.33) into a similarity transformation in the hope that we might be able to get the scalar part to drop out. This strategy turns out to be a good one. We shall see that this approach does not require any restrictive assumptions such as the orthogonality of the quaternion and the vector. In addition, it will turn out to be equivalent to the matrix rotation operator described in (3.5).

$$\mathbf{q}' = \lambda\,\mathbf{q}\,\lambda^{-1} \tag{4.35}$$

Expanding

$$\mathbf{q}' = [\,\lambda_0 \ , \ \bar{\lambda}\,][\ 0 \ , \ \bar{q}\,][\,\lambda_0 \ , \ -\bar{\lambda}\,] = \tag{4.36}$$

$$\left[-\lambda_0\bar{\lambda}\cdot\bar{q} + \lambda_0\bar{q}\cdot\bar{\lambda} + \bar{\lambda}\times\bar{q}\cdot\bar{\lambda} \ , \ +\lambda_0^2\bar{q} + \lambda_0\bar{\lambda}\times\bar{q} + \bar{\lambda}\cdot\bar{q}\bar{\lambda} - \lambda_0\bar{q}\times\bar{\lambda} - \bar{\lambda}\times\bar{q}\times\bar{\lambda} \right]$$

Using the identities $\left[\bar{A}\times\bar{B}\right]\cdot\bar{A} = 0$ and $\bar{A}\times\bar{B}\times\bar{C} = \bar{B}\left[\bar{A}\cdot\bar{C}\right] - \bar{C}\left[\bar{A}\cdot\bar{B}\right]$, with the normalization condition for the unit quaternion λ, viz., (4.29), this becomes

$$\mathbf{q}' = \left[0 \ , \ \left(2\lambda_0^2 - 1\right)\bar{q} + 2\left(\bar{\lambda}\cdot\bar{q}\right)\bar{\lambda} + 2\lambda_0\left(\bar{\lambda}\times\bar{q}\right) \right] \tag{4.37}$$

Thus $\mathbf{q}'$ defined by (4.35) is still a vector or pure quaternion and its length is preserved, as required for a rotation. We need to manipulate these terms so that $\bar{q}$ appears on the right with some operator expression to its left. This vector part can be written in matrix component form by using the following identities

$$\left[\bar{\lambda} \times \bar{q}\right]_i = \epsilon_{ijk}\lambda_j q_k = \tag{4.38}$$

$$\begin{vmatrix} 0 & -\lambda_3 & \lambda_2 \\ \lambda_3 & 0 & -\lambda_1 \\ -\lambda_2 & \lambda_1 & 0 \end{vmatrix} \begin{vmatrix} q_1 \\ q_2 \\ q_3 \end{vmatrix}$$

$$\equiv \tilde{\lambda}\ \bar{q}$$

Also,

$$\left[\left(\bar{\lambda}\cdot\bar{q}\right)\bar{\lambda}\right]_i = \lambda_k q_k \lambda_i = \lambda_i \lambda_k q_k = \left[\left(\bar{\lambda}\ \bar{\lambda}^T\right)\bar{q}\right]_i =$$

$$\begin{vmatrix} \lambda_1\lambda_1 & \lambda_1\lambda_2 & \lambda_1\lambda_3 \\ \lambda_2\lambda_1 & \lambda_2\lambda_2 & \lambda_2\lambda_3 \\ \lambda_3\lambda_1 & \lambda_3\lambda_2 & \lambda_3\lambda_3 \end{vmatrix} \begin{vmatrix} q_1 \\ q_2 \\ q_3 \end{vmatrix} \tag{4.39}$$

where the superscript ^{T} denotes the transpose (interchange of rows and columns). Thus we can write

$$\bar{q}' = \left[\left(2\lambda_0^2 - 1\right)I + 2\lambda\lambda^T + 2\lambda_0\tilde{\lambda}\right]\bar{q} \tag{4.40}$$

$$= 2\begin{vmatrix} \lambda_0^2 + \lambda_1^2 - ½ & \lambda_1\lambda_2 - \lambda_0\lambda_3 & \lambda_1\lambda_3 + \lambda_0\lambda_2 \\ \lambda_1\lambda_2 + \lambda_0\lambda_3 & \lambda_0^2 + \lambda_2^2 - ½ & \lambda_2\lambda_3 - \lambda_0\lambda_1 \\ \lambda_1\lambda_3 - \lambda_0\lambda_2 & \lambda_2\lambda_3 + \lambda_0\lambda_1 & \lambda_0^2 + \lambda_3^2 - ½ \end{vmatrix} \bar{q}$$

$$= 2 \begin{vmatrix} \tfrac{1}{2}[\lambda_0^2+\lambda_1^2-\lambda_2^2-\lambda_3^2] & \lambda_1\lambda_2 - \lambda_0\lambda_3 & \lambda_1\lambda_3 + \lambda_0\lambda_2 \\ \lambda_1\lambda_2 + \lambda_0\lambda_3 & \tfrac{1}{2}[\lambda_0^2-\lambda_1^2+\lambda_2^2-\lambda_3^2] & \lambda_2\lambda_3 - \lambda_0\lambda_1 \\ \lambda_1\lambda_3 - \lambda_0\lambda_2 & \lambda_2\lambda_3 + \lambda_0\lambda_1 & \tfrac{1}{2}[\lambda_0^2-\lambda_1^2-\lambda_2^2+\lambda_3^2] \end{vmatrix} \bar{q}$$

$$\equiv \mathrm{T}\,\bar{q}$$

where I is the identity or unit matrix. The second version of the matrix was obtained using (4.29). The expression (4.35) which involves quaternion operations is equivalent to (4.40) which involves matrix operations.

Alternatively, the quaternion nature of (4.35) may be used more directly as a 4x4 matrix by using the two expressions for quaternion multiplication given in (4.10) and making use of the normalization condition (4.19).

$$\mathbf{q}' = \lambda\, \mathbf{q}\, \lambda^{-1} = \lambda\, \mathbf{q}\, \lambda^{*} = \tag{4.41}$$

$$\begin{vmatrix} +\lambda_0 & -\lambda_1 & -\lambda_2 & -\lambda_3 \\ +\lambda_1 & +\lambda_0 & -\lambda_3 & +\lambda_2 \\ +\lambda_2 & +\lambda_3 & +\lambda_0 & -\lambda_1 \\ +\lambda_3 & -\lambda_2 & +\lambda_1 & +\lambda_0 \end{vmatrix} \begin{vmatrix} +\lambda_0 & +\lambda_1 & +\lambda_2 & +\lambda_3 \\ -\lambda_1 & +\lambda_0 & -\lambda_3 & +\lambda_2 \\ -\lambda_2 & +\lambda_3 & +\lambda_0 & -\lambda_1 \\ -\lambda_3 & -\lambda_2 & +\lambda_1 & +\lambda_0 \end{vmatrix} \begin{vmatrix} 0 \\ q_1 \\ q_2 \\ q_3 \end{vmatrix} =$$

$$\begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & [\lambda_0^2+\lambda_1^2-\lambda_2^2-\lambda_3^2] & 2[\lambda_1\lambda_2 - \lambda_0\lambda_3] & 2[\lambda_1\lambda_3 + \lambda_0\lambda_2] \\ 0 & 2[\lambda_1\lambda_2 + \lambda_0\lambda_3] & [\lambda_0^2-\lambda_1^2+\lambda_2^2-\lambda_3^2] & 2[\lambda_2\lambda_3 - \lambda_0\lambda_1] \\ 0 & 2[\lambda_1\lambda_3 - \lambda_0\lambda_2] & 2[\lambda_2\lambda_3 + \lambda_0\lambda_1] & [\lambda_0^2-\lambda_1^2-\lambda_2^2+\lambda_3^2] \end{vmatrix} \begin{vmatrix} 0 \\ q_1 \\ q_2 \\ q_3 \end{vmatrix}$$

How do we see that (4.40) or (4.41) correspond to (3.5)? Let us start with (4.29).

$$\lambda = \left[\cos(\beta) \; , \; \sin(\beta) \; \hat{\mathbf{n}} \right] \tag{4.29}$$

where

$$\lambda_0 = \cos(\beta)$$

$$\sin(\beta) = \left[1 - \lambda_0^2 \right]^{1/2}$$

$$\hat{\mathbf{n}} = \frac{\hat{\imath}\lambda_1 + \hat{\jmath}\lambda_2 + \hat{k}\lambda_3}{\sin(\beta)}$$

We will establish the connection between (4.35), (4.40) and (3.5) by means of an example rather than a comprehensive and rigorous proof, by dealing with a rotation about a single one of the coordinate axes. Of course, a complete general rotation can be built up by several such component rotations. Take the special case where $\hat{\mathbf{n}}$ lies along the y-axis. Then $\lambda_0 = \cos(\beta)$, $\lambda_2 = \sin(\beta)$ and $\lambda_1 = \lambda_3 = 0$.

The reader can readily verify by substitution into (4.40) that (3.3) will be generated except for the curious fact that the angle ϕ is equal to 2β! This provides insight into the interpretation of the rotation associated with a quaternion. An Euler quaternion generates a rotation about an axis determined by the vector part. The half angle of rotation is determined by the arc tangent of the ratio of the length of the vector part to the scalar part.

The matrix T is the rotation matrix from moving axes to inertial axes. It is numerically identical to the much less symmetric and computationally more complex expression in (3.5). But as simple as this quaternion form of the rotation matrix appears to be, recall it is just the matrix and vector parts of (4.37) under matrix algebra. But (4.37) is identical to (4.35), which is the expression of a rotation of a vector represented as a quaternion with zero scalar part. It is nothing more than a unit quaternion λ multiplied by the vector multiplied by the inverse of the unit quaternion λ, where multiplication means quaternion multiplication. Examine (3.5) and note how cumbersome, and computationally awkward it is. Then examine (4.40) and finally (4.35). How deceptively simple, elegant and computationally efficient (4.40) and (4.35) are! By now the reader should begin to have an appreciation for the power of the quaternion formalism. There is a draw back, however. One can easily intuitively grasp the Euler angles. The four quaternion components are not so easily subject to intuition.

Because of this, it is more convenient to define the initial conditions of a

simulation in terms of Euler angles rather than the quaternions themselves. We can use the Euler angles to generate a rotation matrix **T**. It then remains to find the quaternions that would generate the same rotation matrix. We already know how to generate **T** from the quaternions λ_i. How do we do the inversion, obtain the λ_i from **T**?

The expression (4.35) can be inverted to give the quaternions in terms of the rotation matrix T in (4.40). This inversion can be accomplished as follows. From the diagonal elements of (4.40) and the unit quaternion normalization condition (4.29), we obtain the following four simultaneous equations, where we will be using the trace, Tr [T], which is defined as the sum of the diagonal elements of the matrix T, viz., Tr [T] = $T_{11}+T_{22}+T_{33}$.

$$
\begin{aligned}
T_{11} &= \lambda_0^2 + \lambda_1^2 - \lambda_2^2 - \lambda_3^2 \\
T_{22} &= \lambda_0^2 - \lambda_1^2 + \lambda_2^2 - \lambda_3^2 \\
T_{33} &= \lambda_0^2 - \lambda_1^2 - \lambda_2^2 + \lambda_3^2 \\
1 &= \lambda_0^2 + \lambda_1^2 + \lambda_2^2 + \lambda_3^2
\end{aligned}
\tag{4.42}
$$

These can be solved simultaneously to give the following.

$$
\begin{aligned}
\lambda_0 &= \pm \tfrac{1}{2}[\, 1 + Tr(\mathbf{T})]^{1/2} \\
\lambda_1 &= \pm \tfrac{1}{2}[\, 1 + 2T_{11} - Tr(\mathbf{T})]^{1/2} \\
\lambda_2 &= \pm \tfrac{1}{2}[\, 1 + 2T_{22} - Tr(\mathbf{T})]^{1/2} \\
\lambda_3 &= \pm \tfrac{1}{2}[\, 1 + 2T_{33} - Tr(\mathbf{T})]^{1/2}
\end{aligned}
\tag{4.43}
$$

There is a sign ambiguity to resolve. The chief constraint on the algebraic signs of the λ is that the rotation matrix **T** from which the λ were generated should be regenerated when these λ are substituted in (4.40). Note that no negative matrix elements can arise if all the λ have the same sign, whether positive or negative. Thus the choice of sign is not trivial. The diagonal elements of **T** pose no constraint since the λ occur only as squares in the diagonal. The off-diagonal elements appear as cross terms and are the key to our task. From (4.40), we can take all possible combinations of $\mathbf{T}_{ij} \pm \mathbf{T}_{ji}$, $i \neq j$.

$$
\begin{aligned}
\lambda_1\lambda_2 &= \tfrac{1}{4}[T_{12} + T_{21}] \\
\lambda_0\lambda_3 &= \tfrac{1}{4}[T_{21} - T_{12}] \\
\lambda_1\lambda_3 &= \tfrac{1}{4}[T_{13} + T_{31}] \\
\lambda_0\lambda_2 &= \tfrac{1}{4}[T_{13} - T_{31}] \\
\lambda_2\lambda_3 &= \tfrac{1}{4}[T_{23} + T_{32}] \\
\lambda_0\lambda_1 &= \tfrac{1}{4}[T_{32} - T_{23}]
\end{aligned}
\tag{4.44}
$$

In generating these expressions to investgate the sign ambiguity, we see that (4.44) provides an alternate form for determining the λ, providing that one of the λ_j is known. For example, suppose we obtain λ_0 from (4.43). Then λ_1, λ_2, and λ_3 can be obtained from (4.44) by dividing the appropriate expressions in (4.44) by λ_0. Thus we obtain

$$
\begin{aligned}
\lambda_0 &= \pm\tfrac{1}{2}\left[1 + Tr[T]\right]^{\frac{1}{2}} \\
\lambda_1 &= \frac{1}{4\lambda_0}\left[T_{32} - T_{23}\right] \\
\lambda_2 &= \frac{1}{4\lambda_0}\left[T_{13} - T_{31}\right] \\
\lambda_3 &= \frac{1}{4\lambda_0}\left[T_{21} - T_{12}\right]
\end{aligned}
\tag{4.45}
$$

This hybrid solution combining (4.43) and (4.44) has an unexpected advantage. The sign ambiguity remains only for λ_0. This sign ambiguity is not significant since changing the sign of λ_0 will change the sign of all the λ_j. But the quaternions always appear as products of pairs of quaternions or the square of a quaternion. Thus the rotation matrix is invariant under a sign change for λ_0. See (4.40). Although the sign convention doesn't matter, it is conventionally chosen to be positive.

The above inversion can have numerical problems in computation, however. The λ_j for i=1,2,3, become ill-defined when the λ_0 in the denominator vanishes. This will occur if the trace of the rotation matrix equals -1, which would happen

when $\psi = 0$ and $\phi = -\pi$ radians or 180 degrees. Furthermore, numerical round off errors occur in the immediate neighborhood of this region when using (4.45) for computation.

It is not difficult to avoid this problem since we did not have to start with λ_0 in (4.43). Another choice could have been made before invoking (4.44) for the other 3 λ_i. This could be done by evaluating all four λ_i in (4.43), picking the largest and then using this dominant λ_i to find the other λ in (4.44). We would again obtain results that would be insensitive to sign ambiguity but would now be insensitive to computational round off errors as well by putting us far away from any singularity.

This algorithm is described in Table 1 with two modifications. For consistency, the signs are examined at the end of the algorithm. If λ_0 is negative, the signs of all λ_i are reversed to keep λ_0 positive, according to our convention. We have already seen that changing the overall sign of all four quaternions has no effect on the rotation matrix. Secondly, evaluating all four quaternions using (4.43) is computationally wasteful. It is sufficient to compare the trace and the three diagonal elements of **T** to select the dominant quaternion. The complete algorithm is shown in Table 1.

In order to use the quaternions we must have a scheme for determining the time rate of change of the quaternions so that these rates can be integrated to provide updated quaternions as the system evolves in time. These would be expressions that play a role completely analogous to (3.16) through (3.18) for the Euler angles. This derivation could have been accomplished by standard matrix algebraic techniques. However, the derivation would have been exceedingly long and tedious. Using quaternions, it is rather simple and straight forward.

Since $\dot{\mathbf{q}}$ and $\dot{\mathbf{q}}'$ are vectors or pure quaternions, they obey the transformation law given in (4.35),

$$\mathbf{q}' = \lambda\, \mathbf{q}\, \lambda^{-1} \qquad\qquad \mathbf{q} = \lambda^{-1}\, \mathbf{q}'\, \lambda$$
$$\dot{\mathbf{q}}' = \lambda\, \dot{\mathbf{q}}\, \lambda^{-1} \qquad\qquad \dot{\mathbf{q}} = \lambda^{-1}\, \dot{\mathbf{q}}'\, \lambda \tag{4.46}$$

Now we wish to take the time derivative of **q** in the moving coordinate frame. As is well-known, a term must be added due to the angular velocity $\overline{\Omega}$ of the rotating coordinate frame. Taking the derivative

Table 1. Obtaining the Quaternions from the Rotation Matrix T

- **Define T_{00} = Tr [T].**
- **Compare T_{ii} where 1 = 0,1,2,3.**

  Find the dominant quaternion (most positive or least negative.)
  The index of the dominant T_{ii} determines the dominant λ_i to be used in (4.43).

- **Determine the other three λ_j, $j \neq i$ in (4.44).**

  The four cases, with the dominant λ first are

$$\lambda_0 = \pm \tfrac{1}{2}[\, 1 + Tr(\mathbf{T})]^{1/2} \qquad \lambda_1 = \pm \tfrac{1}{2}[\, 1 + 2T_{11} - Tr(\mathbf{T})]^{1/2}$$

$$\lambda_1 = \frac{1}{4\lambda_0}\left[T_{32} - T_{23}\right] \qquad \lambda_0 = \frac{1}{4\lambda_1}\left[T_{32} - T_{23}\right]$$

$$\lambda_2 = \frac{1}{4\lambda_0}\left[T_{13} - T_{31}\right] \qquad \lambda_2 = \frac{1}{4\lambda_1}\left[T_{12} + T_{21}\right]$$

$$\lambda_3 = \frac{1}{4\lambda_0}\left[T_{21} - T_{12}\right] \qquad \lambda_3 = \frac{1}{4\lambda_1}\left[T_{13} + T_{31}\right]$$

$$\lambda_2 = \pm \tfrac{1}{2}[\, 1 + 2T_{22} - Tr(\mathbf{T})]^{1/2} \qquad \lambda_3 = \pm \tfrac{1}{2}[\, 1 + 2T_{33} + Tr(\mathbf{T})]^{1/2}$$

$$\lambda_0 = \frac{1}{4\lambda_2}\left[T_{13} - T_{31}\right] \qquad \lambda_0 = \frac{1}{4\lambda_3}\left[T_{21} - T_{12}\right]$$

$$\lambda_1 = \frac{1}{4\lambda_2}\left[T_{12} + T_{21}\right] \qquad \lambda_1 = \frac{1}{4\lambda_3}\left[T_{13} + T_{31}\right]$$

$$\lambda_3 = \frac{1}{4\lambda_2}\left[T_{23} + T_{32}\right] \qquad \lambda_2 = \frac{1}{4\lambda_3}\left[T_{23} + T_{32}\right]$$

- **Examine the algebraic sign of λ_0. If negative, change the sign of all four λ_j.**

$$\frac{d\mathbf{q}}{dt} = \frac{d}{dt}[\lambda^{-1}\mathbf{q}'\lambda] + [0, \bar{\Omega}\mathrm{x}\bar{q}]$$

$$= \dot{\lambda}^{-1}\mathbf{q}'\lambda + \lambda^{-1}\dot{\mathbf{q}}'\lambda + \lambda^{-1}\mathbf{q}'\dot{\lambda} + [0, \bar{\Omega}\mathrm{x}\bar{q}] \tag{4.47}$$

All these terms are pure quaternions or vectors. Since

$$\lambda^{-1}\dot{\mathbf{q}}'\lambda = \lambda^{-1}\lambda\dot{\mathbf{q}}\lambda^{-1}\lambda = \dot{\mathbf{q}} \tag{4.48}$$

we conclude that

$$\dot{\lambda}^{-1}\mathbf{q}'\lambda + \lambda^{-1}\mathbf{q}'\dot{\lambda} = -[0, \bar{\Omega}\mathrm{x}\bar{\mathbf{q}}] = +[0, \bar{\mathbf{q}}\mathrm{x}\bar{\Omega}]$$

or (4.49)

$$\dot{\lambda}^{-1}\lambda\mathbf{q}\lambda^{-1}\lambda + \lambda^{-1}\lambda\mathbf{q}\lambda^{-1}\dot{\lambda} = +[0, \bar{\mathbf{q}}\mathrm{x}\bar{\Omega}]$$

$$= \dot{\lambda}^{-1}\lambda\mathbf{q} + \mathbf{q}\lambda^{-1}\dot{\lambda}$$

Differentiating $\lambda^{-1}\lambda = \overline{[1,0]}$ gives $\dot{\lambda}^{-1}\lambda = -\lambda^{-1}\dot{\lambda}$. Thus from (4.7)

$$[0, \bar{\mathbf{q}}\mathrm{x}\bar{\Omega}] = \mathbf{q}\lambda^{-1}\dot{\lambda} - \lambda^{-1}\dot{\lambda}\mathbf{q} = 2[0, \bar{\mathbf{q}}\mathrm{x}\overline{[\lambda^{-1}\dot{\lambda}]}] \tag{4.50}$$

Recall from (4.28) that $\lambda^{-1}\dot{\lambda}$ is a pure quaternion or vector. Thus

$$\frac{1}{2}\Omega = \lambda^{-1}\dot{\lambda} \tag{4.51}$$

We use the i, j, k notation described in (4.1) through (4.6), since this formalism makes it easier to group terms and find a matrix operator equivalent. We obtain

$$2\lambda^{-1}\dot{\lambda} = \Omega \equiv$$

$$[0, \bar{\Omega}] = [0 + i\Omega_1 + j\Omega_2 + k\Omega_3] = \tag{4.52}$$

$$2[\lambda_0 - i\lambda_1 - j\lambda_2 - k\lambda_3][\dot{\lambda}_0 + i\dot{\lambda}_1 + j\dot{\lambda}_2 + k\dot{\lambda}_3]$$

$$2\left[\lambda_0\dot{\lambda}_0+\lambda_1\dot{\lambda}_1+\lambda_2\dot{\lambda}_2+\lambda_3\dot{\lambda}_3\right] + 2i\left[-\lambda_1\dot{\lambda}_0+\lambda_0\dot{\lambda}_1+\lambda_3\dot{\lambda}_2-\lambda_2\dot{\lambda}_3\right] +$$

$$2j\left[-\lambda_2\dot{\lambda}_0-\lambda_3\dot{\lambda}_1+\lambda_0\dot{\lambda}_2+\lambda_1\dot{\lambda}_3\right] + 2k\left[-\lambda_3\dot{\lambda}_0+\lambda_2\dot{\lambda}_1-\lambda_1\dot{\lambda}_2+\lambda_0\dot{\lambda}_3\right]$$

This may be expressed into the matrix form (recall equation 4.10)

$$\begin{vmatrix} 0 \\ \Omega_x \\ \Omega_y \\ \Omega_z \end{vmatrix} = 2 \begin{vmatrix} +\lambda_0 & +\lambda_1 & +\lambda_2 & +\lambda_3 \\ -\lambda_1 & +\lambda_0 & +\lambda_3 & -\lambda_2 \\ -\lambda_2 & -\lambda_3 & +\lambda_0 & +\lambda_1 \\ -\lambda_3 & +\lambda_2 & -\lambda_1 & +\lambda_0 \end{vmatrix} \begin{vmatrix} \dot{\lambda}_0 \\ \dot{\lambda}_1 \\ \dot{\lambda}_2 \\ \dot{\lambda}_3 \end{vmatrix} \tag{4.53}$$

The top element in the column matrix for the angular velocity can be seen to vanish by taking the time derivative of the normalization condition on λ, viz., (4.29). Similarly, the first line of (4.52) can be quaternion-multiplied from the right by λ to yield

$$\dot{\lambda} = ½\lambda\Omega =$$

$$½[\lambda_0 + i\lambda_1 + j\lambda_2 + k\lambda_3][0 + i\Omega_x + j\Omega_y + k\Omega_z] = \tag{4.54}$$

$$½\,[\,\lambda_0 0-\lambda_1\Omega_x-\lambda_2\Omega_y-\lambda_3\Omega_z] + ½i\,[\lambda_1 0+\lambda_0\Omega_x-\lambda_3\Omega_y+\lambda_2\Omega_z\,]$$

$$+\ ½j\,[\,\lambda_2 0+\lambda_3\Omega_x+\lambda_0\Omega_y-\lambda_1\Omega_z] + ½k\,[\lambda_3 0-\lambda_2\Omega_x+\lambda_1\Omega_y+\lambda_0\Omega_z\,]$$

This may be organized into the matrix form using (4.10)

$$\begin{vmatrix} \dot{\lambda}_0 \\ \dot{\lambda}_1 \\ \dot{\lambda}_2 \\ \dot{\lambda}_3 \end{vmatrix} = ½ \begin{vmatrix} +\lambda_0 & -\lambda_1 & -\lambda_2 & -\lambda_3 \\ +\lambda_1 & +\lambda_0 & -\lambda_3 & +\lambda_2 \\ +\lambda_2 & +\lambda_3 & +\lambda_0 & -\lambda_1 \\ +\lambda_3 & -\lambda_2 & +\lambda_1 & +\lambda_0 \end{vmatrix} \begin{vmatrix} 0 \\ \Omega_x \\ \Omega_y \\ \Omega_z \end{vmatrix} \tag{4.55}$$

See equation (4.10) Note that both (4.53) and (4.55) are free of singularities, unlike their analogs (3.19) and (3.20). Both matrices (4.53) and (4.55) are equal to a unit matrix times λ_o added to an antisymmetric matrix. These matrices are inverses of one another as well as transposes of one another. Hence they are orthogonal and preserve the length of the vectors they operate on. Thus the vectors (λ_1 λ_1 λ_2 λ_3) and (0 Ω_x Ω_y Ω_z) are related by a rotation in 4-space with the latter having only a three-vector part. Compare these elegant properties and the simplicity and low computational burden associated with the use of these equations versus (3.19) and (3.20). If the above derivation is not simple enough for you, the matrix in (4.55) may be obtained by realizing it is the inverse of the matrix in (4.53), and can be obtained by substituting into the matrix in (4.53) the inverse of the unit quaternion λ. But from (4.22) we realize that the inverse of a unit quaternion is obtained merely by changing the sign of the last three ("vector-like") components of the quaternion. Thus all that needs to be done to invert this matrix is to change the sign of the off-diagonal elements. There is something here for everyone: the mathematician, the physicist, the engineer and the programmer.

Since quaternions are not as intuitive as Euler angles, it is sometimes desirable to move back and forth between the Euler angle and quaternion representations. Going from Euler angle to quaternions representation can be achieved by using the Euler angles to evaluate the transformation matrix using equation (3.5). The transformation matrix T can then be put into (4.41) to obtain the quaternions. Note that after (4.41), it was noted that the sign for λ_o is not important. Why this is so becomes apparent shortly. From the first row and last column of (3.5), and (4.40)

$$\begin{aligned}
\sin\theta &= -T_{31} = 2[\lambda_0\lambda_2 - \lambda_1\lambda_3] && -\pi/2 \le \theta \le \pi/2 \\
\tan\psi &= \frac{T_{21}}{T_{11}} = \frac{2[\lambda_1\lambda_2 + \lambda_0\lambda_3]}{\lambda_0^2 + \lambda_1^2 - \lambda_2^2 - \lambda_3^2} && -\pi < \psi \le \pi \\
\tan\phi &= \frac{T_{32}}{T_{33}} = \frac{2[\lambda_2\lambda_3 + \lambda_0\lambda_1]}{\lambda_0^2 - \lambda_1^2 - \lambda_2^2 + \lambda_3^2} && 0 \le \phi < 2\pi .
\end{aligned} \tag{4.56}$$

Note that theta can only be defined in this way over a range π where the arcsin is defined, viz. between $+\pi/2$ and $-\pi/2$. However, ψ and ϕ can be found over a full 360 degrees or 2 π radians by taking into account the sign of the numerator and the sign of the denominator of the last two expressions. See the algorithm programmed in Table 2. Note further that either sign selection for λ_o made in (4.41) will yield back the same Euler angles in the above expression. Thus the sign choice is arbitrary.

Table 2. Evaluation of ARCTAN(A,B) Over All Four Quadrants

IF B>0	$\tan^{-1}(\mathbf{A}/\mathbf{B})$
IF B=0, A>0	$\pi/2$
IF B=0, A<0	$-\pi/2$
IF B<0	$\pi + \tan^{-1}(\mathbf{A}/\mathbf{B})$

The denominator in two of the expressions given in (4.56) can vanish when ψ or θ are at $\pm\pi/2$. The algorithm in Table 2 will treat this correctly.

We shall see in the next chapter that plane-fixed coordinates will require $\phi = 0$. When applied to the expression for $\tan \phi$ in (4.56), this constraint will further require

$$\lambda_2\lambda_3 + \lambda_0\lambda_1 = 0 \tag{4.57}$$

A note of caution should be addressed, viz., the necessity for renormalization. Round off error in a computer simulation gradually causes the normalization condition (4.20) or (4.29) to fail. Thus the quaternions need to be regularly renormalized in a simulation.

5 EQUATIONS OF MOTION

In this section, the rigid body equations of motion will be developed for three types of coordinate systems: body-fixed, aeroballistic and plane-fixed. Body-fixed coordinates rotate (roll, pitch, yaw) with the body of a projectile. Hence the angular velocity Ω of the coordinate axes is equal to the angular velocity ω of the projectile body. See (3.13) through (3.15). For *body-fixed coordinates*

$$
\begin{aligned}
\Omega_x &= \omega_x \equiv P = -\dot{\psi} \sin \theta + \dot{\phi} \\
\Omega_y &= \omega_y \equiv Q = \dot{\psi} \sin \phi \cos \theta + \dot{\theta} \cos \phi \\
\Omega_z &= \omega_z \equiv R = \dot{\psi} \cos \phi \cos \theta - \dot{\theta} \sin \phi
\end{aligned}
\tag{5.1}
$$

Recall that in aerodynamics the components of the angular velocity of the projectile body $\overline{\omega}$ are conventionally denoted by P, Q and R. These coordinates are the natural choice for guided projectiles since the seeker and sensor outputs, actuator parameters, and so forth are most simply and naturally expressed in body coordinates. However, for spin stabilized projectiles, a very small integration time step is required. Otherwise the coordinate system will roll through too great an angle during the time step and smear the direction of forces such as gravity. To deal with this, plane-fixed coordinates are utilized. These coordinates pitch and yaw with the projectile but do not roll with it. In particular, one axis is constrained to remaining in a single plane. In our case, the y-axis will be constrained to the horizontal plane. See equations (3.22) and following. This implies that the roll Euler angle of the *plane-fixed frame* satisfies the relations

$$
d\phi/dt = 0 \ , \quad \phi = 0 \tag{5.2}
$$

Thus, although the projectile is rolling, the fixed plane coordinate system is not rolling, in the sense that ϕ vanishes for the fixed plane axes. However, the x component of the angular velocity of the frame, viz. Ω_x does not vanish and Ω_x of the frame does not equal P (the x component of the projectile angular velocity). These relations may be found in equations (3.26) through (3.28).

$$
\begin{aligned}
\Omega_x &\equiv -\dot{\psi} \sin\theta = -R \tan\theta \\
\Omega_y &\equiv Q = \dot{\theta} \\
\Omega_z &\equiv R = \dot{\psi} \cos\theta
\end{aligned}
\tag{5.3}
$$

Comparing this expression for Ω_x with (5.1), we see that $\dot{\phi}$ is the roll rate of the projectile with respect to plane-fixed coordinate frame. (5.1) can be reconstructed from (5.3) by adding $\dot{\phi}$ to (5.3) and rotating with (3.3). This expression in useful if the Euler angle representation is used. With the quaternion representation for plane-fixed coordinates, we make use of (3.5), (4.40) and $\phi = 0$ to obtain

$$\tan\theta = \frac{-T_{31}}{T_{33}} = \frac{-2\left[\lambda_1\lambda_3 - \lambda_0\lambda_2\right]}{\lambda_0^2 - \lambda_1^2 - \lambda_2^2 + \lambda_3^2} \tag{5.4}$$

Thus

$$\Omega_x = \frac{2R\left[\lambda_1\lambda_3 - \lambda_0\lambda_2\right]}{\lambda_0^2 - \lambda_1^2 - \lambda_2^2 + \lambda_3^2} \tag{5.5}$$

The denominators in (5.4) and (5.5) can vanish, corresponding to $\theta = \pm\pi/2$. In defining plane-fixed coordinates, we had to impose Euler angle type algebra on the quaternions and have corrupted them with Euler angle type singularities. Thus, *quaternions in plane-fixed coordinates should not be used in vertical trajectories. Use body-fixed coordinates instead.* This is not burdensome computationally since the direction of gravity is along the axis of roll. Finally, we could choose $\Omega_x = 0$ in (5.1) for *aeroballistic coordinates*. This choice will simplify equations of motion[1].

In the following development, results will be derived using the term Ω_x. This will allow all three coordinate formalisms to be developed simultaneously. At the end, the results can be specialized to one or the other by letting Ω_x equal P for the body-fixed case or equal zero for the aeroballistic ("zero P") case or equal $-R\tan\theta$ (or its quaternion equivalent - see (5.3) and (5.5)). $\Omega_y = Q$ and $\Omega_z = R$ for all three cases. From (3.13) we see that the definitions for aeroballistic and body-fixed frames are not equivalent but become the same in the limit of small θ or small $d\psi/dt$. The advantage of choosing plane-fixed coordinates for a non-rolling system is that it has no y component of gravity in a flat earth model, thus eliminating the possibility of gravity smearing due to roll during integration of the equations of motion. (This will be shown in equation (5.10) below.)

In summary, the body-fixed coordinates roll, pitch and yaw with the projectile and act as if physically attached to the projectile. Plane-fixed coordinates pitch and yaw with the projectile but do not roll with it. The y-axis is constrained to move in the horizontal plane. See the discussion for equations (3.22) to (3.33). The Euler angle rotation matrix for the plane-fixed case can be obtained from the body-fixed matrix (3.5) by letting $\phi = 0$. Equivalently, (3.3) is replaced by the unit identity matrix.

---

[1] Vaughn, Harold R., "A detailed Development of the Tricyclic Theory," Sandia Laboratories, SC-M-67-2933, Albuquerque, NM, 1968.

The form of Newton's law F = dmv/dt is valid only in inertial (i.e., non-accelerating) coordinate frames. If the coordinate system is rotating, Newton's law will not be valid because a rotation is an acceleration. However, the law can be amended for rotational frames. As is well-known, Newton's law for linear accelerations and forces in a rotating frame takes the form (the superscript M denotes the moving frame, body-fixed, plane-fixed or aeroballistic)[1,2,3,4]

$$\bar{F} + m\bar{g}^{M} = \frac{d}{dt}\overline{mV} + \bar{\Omega} x \overline{mV} = m\dot{\bar{V}} + m\,\bar{\Omega} x \bar{V} \qquad (5.6)$$

$\bar{F}$ contains applied forces such as thrust and aerodynamic forces. Since the coordinate system is non-inertial, it also contains "fictitious" terms such as centrifugal and Coriolis "forces". Derivatives of inertial properties such as mass will be omitted in this development[5]. Defining V_x = U, V_y = V, and V_z = W, the components of the above vector equation is now

$$\begin{aligned} F_x + mg_x^{M} &= m\dot{U} + m\,[QW - RV] \\ F_y + mg_y^{M} &= m\dot{V} + m\,[RU - \Omega_x W] \\ F_z + mg_z^{M} &= m\dot{W} + m\,[\Omega_x V - QU] \end{aligned} \qquad (5.7)$$

Rearranging

$$\dot{U} = \frac{F_x}{m} + g_x^{M} - QW + RV$$

---

1 John H. Blakelock, "Automatic Control of Aircraft and Missiles," John Wiley and Sons, New York, 1965.

2 Keith K. Symon, "Mechanics," Addison-Wesley, Reading, Mass, 1960.

3 Goldstein, Herbert, "Classical Mechanics", p 136, Addison Wesley, Reading, Mass, 1959.

4 Landau, L. D., and E. M. Lifshitz, "Classical Mechanics", p 128, Addison Wesley, Mass, 1960.

5 The thrust of a reaction engine that is measured in a test stand already contains the effects of the rate of change of the mass. Since this information is usually available for input into a simulation rather than nozzle pressures, derivatives of the mass do not appear in the equations of motion. However, if the thrust is to be reconstructed from pressure measurements at the nozzle, mass derivative terms and the velocity of the exhaust gases would have to be taken into account in the equations of motion. This will be discussed in detail in a future report.

$$\dot{V} = \frac{F_y}{m} + g_y^M - RU + \Omega_x W \tag{5.8}$$

$$\dot{W} = \frac{F_z}{m} + g_z^M - \Omega_x V + QU$$

The components for $\bar{g}$ can be obtained by multiplying the gravity vector in the earth frame by the appropriate T matrix. For the special case for a flat earth, $\bar{g}$ only has a vertical or z component pointing downward. (Coriolis and centripetal acceleration corrections should be made since a flat earth is not really inertial since the earth rotates. If distances flown and time of flight are short, these corrections are negligible.)

We need the inertial to body transformation (from the inverse of (4.40)) and the representation of the gravity vector in a flat earth as (0 , 0 , g)

$$g_x^M = 2[\lambda_1\lambda_3 - \lambda_0\lambda_2]g$$

$$g_y^M = 2[\lambda_2\lambda_3 + \lambda_0\lambda_1]g \qquad \text{for flat earth} \tag{5.9}$$

$$g_z^M = [\lambda_0^2 - \lambda_1^2 - \lambda_2^2 + \lambda_3^2]g$$

If the user wishes, the components for $\bar{g}$ in the aeroballistic, plane-fixed or body-fixed frames for a flat earth could be obtained in terms of the Euler angles instead of quaternions by using the expression for $T^{-1} = T^T$ in (3.5) instead of (4.40).

$$g_x^M = -g \sin\theta$$

$$g_y^M = +g \sin\phi \cos\theta \qquad \text{for flat earth} \tag{5.10}$$

$$g_x^M = +g \cos\phi \cos\theta$$

Note that there is no plane-fixed y-axis component of gravity since sin ϕ for plane-fixed coordinates. advantage of the definition of plane-fixed coordinates adopted in this development. It would not be true if $\Omega_x = 0$ was chosen. In general, if $\bar{g}$ has x or y components in inertial coordinates, the full rotation matrix T would have to be used. See (3.5) and (4.40).

Recalling that Ω is the angular velocity of the coordinate frame with respect to the inertial frame, ω is the angular velocity of the body itself with respect to the inertial frame, and the moment of inertia tensor is symmetric, i.e., $I_{ij} = I_{ji}$; Newton's laws for angular velocities and moments are of the form

$$\overline{M} = \dot{\overline{L}} + \overline{\Omega} x \overline{L} = I\dot{\omega} + \Omega x [I\omega] \tag{5.11}$$

$$\begin{vmatrix} M_x \\ M_y \\ M_z \end{vmatrix} = \begin{vmatrix} +I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & +I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & +I_{zz} \end{vmatrix} \begin{vmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{vmatrix}$$

Time derivative of the angular velocity of the body

Angular velocity of coordinate system

$$+ \begin{vmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{vmatrix} \text{ x } \begin{vmatrix} +I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & +I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & +I_{zz} \end{vmatrix} \begin{vmatrix} \omega_x \\ \omega_y \\ \omega_z \end{vmatrix}$$

cross product Angular velocity of the body

In aerodynamic conventional notation,

$$
\begin{aligned}
M_x &= \mathrm{L} \qquad & \omega_x &= \mathrm{P} \\
M_y &= \mathrm{M} \qquad & \omega_y &= \mathrm{Q} \\
M_z &= \mathrm{N} \qquad & \omega_z &= \mathrm{R}
\end{aligned}
\tag{5.12}
$$

Body-fixed, aeroballistic and plane-fixed coordinates can be developed simultaneously by writing $\Omega_x = \mathrm{P}$ for body-fixed, $\Omega_x = 0$ for aeroballistic and by writing for plane-fixed coördinates

$$
\begin{aligned}
\Omega_x &= 2R\,[\lambda_1\lambda_3 - \lambda_0\lambda_2] \,/\, [\lambda_0^2 - \lambda_1^2 - \lambda_2^2 + \lambda_3^2] \\
&= -R\,\tan\theta
\end{aligned}
\tag{5.13}
$$

See (5.2) or (5.5).

$$
\left|\begin{matrix} L \\ M \\ N \end{matrix}\right| = \left|\begin{matrix} +I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & +I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & +I_{zz} \end{matrix}\right| \left|\begin{matrix} \dot{P} \\ \dot{Q} \\ \dot{R} \end{matrix}\right|
$$

$$
+ \left|\begin{matrix} \Omega_x \\ Q \\ R \end{matrix}\right| \times \left|\begin{matrix} +I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & +I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & +I_{zz} \end{matrix}\right| \left|\begin{matrix} P \\ Q \\ R \end{matrix}\right|
\tag{5.14}
$$

$$
= \left| \begin{matrix} +I_{xx}\dot{P} & -I_{xy}\dot{Q} & -I_{xz}\dot{R} \\ -I_{yx}\dot{P} & +I_{yy}\dot{Q} & -I_{yz}\dot{R} \\ -I_{zx}\dot{P} & -I_{zy}\dot{Q} & +I_{zz}\dot{R} \end{matrix} \right| +
$$

$$
\left| \begin{matrix} Q\,[\,-I_{zx}P \; -I_{zy}Q \; +I_{zz}R\,] - R\,[\,-I_{yx}P \; +I_{yy}Q \; -I_{yz}R\,] \\ R\,[\,+I_{xx}P \; -I_{xy}Q \; -I_{xz}R\,] - \Omega_x\,[\,-I_{zx}P \; -I_{zy}Q \; +I_{zz}R\,] \\ \Omega_x\,[\,-I_{yx}P \; +I_{yy}Q \; -I_{yz}R\,] - Q\,[\,+I_{xx}P \; -I_{xy}Q \; -I_{xz}R\,] \end{matrix} \right|
$$

This may be written

$$
L = I_{xx}\dot{P} \tag{5.15}
$$

$$
+ \left[I_{zz} - I_{yy}\right]QR
$$
<- Vanishes with axial symmetry

$$
+I_{yz}\left[R^2 - Q^2\right] + I_{xz}\left[-QP - \dot{R}\right] + I_{xy}\left[RP - \dot{Q}\right]
$$
<- Vanish if products of inertia vanish

$$
M = I_{xx}PR + I_{yy}\dot{Q} - I_{zz}R\Omega_x
$$

$$
+I_{yz}\left[Q\Omega_x - \dot{R}\right] + I_{xz}\left[\Omega_x P - R^2\right] + I_{xy}\left[-QR - \dot{P}\right]
$$
<- Vanish if products of inertia vanish

$$
N = -I_{xx}PQ + I_{yy}\Omega_x Q + I_{zz}\dot{R}
$$

$$
+I_{yz}\left[-\Omega_x R - \dot{Q}\right] + I_{xz}\left[QR - \dot{P}\right] + I_{xy}\left[Q^2 - \Omega_x P\right]
$$
<- Vanish if products of inertia vanish

If Ω_x is replaced by P, equations (5.14) or (5.15) are the general equations of motion for the angular velocity components of a rigid body in body-fixed coordinates. For convenience in programming, terms that vanish when the products of inertia vanish have been grouped together. Body-fixed coordinates are appropriate for unsymmetric bodies and for guided projectiles, since the sensors and control system are naturally described in body-fixed coordinates that roll with the body. Note that the components of the moment of inertia tensor **I** will not change due to the rotation of the body-fixed coordinates.

For axially-symmetric spin-stabilized projectiles, plane-fixed coordinates are more appropriate. If body-fixed coordinates are used with rapidly spinning spin stabilized projectiles, the integration time step in a 6 degree-of-freedom simulation is driven to be very small, increasing the simulation run time. This is necessary to keep the roll angle during the integration time step small. This avoids smearing gravity in the expressions for $\dot{x}$, $\dot{y}$ and $\dot{z}$. See (5.7) or (5.8). With plane-fixed coordinates and axial symmetry, the products of inertia I_{xy}, I_{yz}, and I_{xz} vanish, and $I_{yy} = I_{zz}$. Ω_x is replaced using (5.13) in (5.14) or (5.15). Note that the components of the moment of inertia tensor **I** would generally vary with time if the projectile were rotating but the frame were not. Not only would this be a complication but it would drive down the integration time step and increase integration time. Similarly for the aerodynamics. Since the motivation for plane-fixed coordinates is greater computational speed, it is pointless to eliminate gravitational smear and substitute inertial or aerodynamic smear. Thus axial symmetry in mass properties and aerodynamics is generally assumed.

After one more result is obtained, these formulae will be collected in Tables 3 to 8. From (4.55), using the appropriate expression for Ω_x, we can write down the time derivatives of the quaternions for these frames.

For body-fixed,

$$\dot{\lambda}_0 = ½\left[-P\lambda_1 - Q\lambda_2 - R\lambda_3\right]$$

$$\dot{\lambda}_1 = ½\left[+P\lambda_0 - Q\lambda_3 + R\lambda_2\right]$$

$$\dot{\lambda}_2 = ½\left[+P\lambda_3 + Q\lambda_0 - R\lambda_1\right] \tag{5.16}$$

$$\dot{\lambda}_3 = ½\left[-P\lambda_2 + Q\lambda_1 + R\lambda_0\right]$$

For plane-fixed, using (4.55) and (5.5)

$$\dot{\lambda}_0 = ½\left\{-2R\lambda_1\left[\frac{\lambda_1\lambda_3-\lambda_0\lambda_2}{[\lambda_0^2-\lambda_1^2-\lambda_2^2+\lambda_3^2]}\right] - Q\lambda_2 - R\lambda_3\right\}$$

$$= -½\left\{\lambda_2 Q + \frac{\lambda_3 R}{\left[\lambda_0^2-\lambda_1^2-\lambda_2^2+\lambda_3^2\right]}\right\}$$

$$\dot{\lambda}_1 = ½\left\{+2R\lambda_0\left[\frac{\lambda_1\lambda_3-\lambda_0\lambda_2}{[\lambda_0^2-\lambda_1^2-\lambda_2^2+\lambda_3^2]}\right] - Q\lambda_3 + R\lambda_2\right\} \tag{5.17}$$

$$= -½\left\{\lambda_3 Q + \frac{\lambda_2 R}{\left[\lambda_0^2-\lambda_1^2-\lambda_2^2+\lambda_3^2\right]}\right\}$$

$$\dot{\lambda}_2 = ½\left\{+2R\lambda_3\left[\frac{\lambda_1\lambda_3-\lambda_0\lambda_2}{[\lambda_0^2-\lambda_1^2-\lambda_2^2+\lambda_3^2]}\right] + Q\lambda_0 - R\lambda_1\right\}$$

$$= +½\left\{\lambda_0 Q + \frac{\lambda_1 R}{\left[\lambda_0^2-\lambda_1^2-\lambda_2^2+\lambda_3^2\right]}\right\}$$

$$\dot{\lambda}_3 = ½\left\{-2R\lambda_2\left[\frac{\lambda_1\lambda_3-\lambda_0\lambda_2}{[\lambda_0^2-\lambda_1^2-\lambda_2^2+\lambda_3^2]}\right] + Q\lambda_1 + R\lambda_0\right\}$$

$$= +½\left\{\lambda_1 Q + \frac{\lambda_0 R}{\left[\lambda_0^2-\lambda_1^2-\lambda_2^2+\lambda_3^2\right]}\right\}$$

A singularity will occur if the denominator vanishes. From (3.22) and (4.40), this denominator is just $\mathbf{T}_{33} = \cos(\theta)$, which vanishes for $\theta = \pm\pi/2$. This is the same singularity discussed after (5.5). It is not advisable to use plane-fixed coordinates for near vertical trajectories. Use body-fixed coordinates instead. For vertical trajectories, gravity smearing due to roll rate is not the problem it is for other trajectories.

All the quaternion and Euler angle results for the linear and angular equations of motion are collected in the following tables for body-fixed and for plane-fixed coordinates respectively.

For completeness, we note that the aeroballistic frame equations of motion can be obtained by letting Ω_x vanish in (5.8) and (5.15), and letting Ω_x vanish in (4.55) or P vanish in (5.16). As with plane-fixed coordinates, we further simplify by neglecting the products of inertia. This form will be found in Table 5. Further simplification results from assuming axial symmetry and constant mass. See Tables 5 and 6.

Table 3. Body-Fixed Equations

$$\dot{U} = \frac{F_x}{m} + g_x^M - QW + RV$$

$$\dot{V} = \frac{F_y}{m} + g_y^M - RU + PW \qquad (5.1), (5.8)$$

$$\dot{W} = \frac{F_z}{m} + g_z^M - PV + QU$$

The components of gravitational acceleration are obtained using T^{-1}.
For the flat earth approximation, use (5.9) and (5.10).

$$L = I_{xx}\dot{P}$$

$$+ \left[I_{zz} - I_{yy}\right]QR \quad \text{<- Vanishes for axial symmetry} \qquad (5.15)$$

$$+ I_{yz}\left[R^2 - Q^2\right] + I_{xz}\left[-QP - \dot{R}\right] + I_{xy}\left[RP - \dot{Q}\right] \quad \text{<- Vanish if products of inertia vanish}$$

$$M = +I_{yy}\dot{Q} + \left[I_{xx} - I_{zz}\right]RP$$

$$+ I_{yz}\left[QP - \dot{R}\right] + I_{xz}\left[P^2 - R^2\right] + I_{xy}\left[-QR - \dot{P}\right] \quad \text{<- Vanish if products of inertia vanish}$$

$$N = \left[I_{yy} - I_{xx}\right]PQ + I_{zz}\dot{R}$$

$$+ I_{yz}\left[-PR - \dot{Q}\right] + I_{xz}\left[QR - \dot{P}\right] + I_{xy}\left[Q^2 - P^2\right] \quad \text{<- Vanish if products of inertia vanish}$$

Table 4. Time Development of the Body-Fixed Transformation Matrix Parameters

$$\begin{aligned}
\dot{\lambda}_0 &= ½\left[-P\lambda_1 - Q\lambda_2 - R\lambda_3\right] \\
\dot{\lambda}_1 &= ½\left[+P\lambda_0 - Q\lambda_3 + R\lambda_2\right] \\
\dot{\lambda}_2 &= ½\left[+P\lambda_3 + Q\lambda_0 - R\lambda_1\right] \\
\dot{\lambda}_3 &= ½\left[-P\lambda_2 + Q\lambda_1 + R\lambda_0\right]
\end{aligned} \tag{5.16}$$

or

$$\dot{\psi} = \frac{\left(Q\ sin(\phi) + R\ cos(\phi)\right)}{cos(\theta)} \tag{3.16}$$

$$\dot{\phi} = P + \left(Q\ sin(\phi) + R\ cos(\phi)\right)\ tan(\theta) \tag{3.17}$$

$$\dot{\theta} = Q\ cos(\phi) - R\ sin(\phi) \tag{3.18}$$

(The expressions (3.16) and (3.17) have a singularity at $\pm\ \pi/2$. This singularity does not occur in the quaternion expression (5.16). See discussion in text.)

Table 5. Plane-Fixed Equations

(Axial Symmetry, $I_{yy} = I_{zz} \equiv I_t$, products of inertia vanish)

$$\dot{U} = \frac{F_x}{m} + g_x^M - QW + RV$$

$$\dot{V} = \frac{F_y}{m} + g_y^M - RU + \Omega_x W \tag{5.8}$$

$$\dot{W} = \frac{F_z}{m} + g_z^M - \Omega_x V + QU$$

The components of gravitational acceleration are obtained using T^{-1}. For the flat earth approximation, use (5.9) and (5.10).

$$L = I_{xx} \dot{P}$$

$$M = I_t \dot{Q} + I_{xx} PR - I_t R \Omega_x \tag{5.15}$$

$$N = I_t \dot{R} - I_{xx} QP + I_t \Omega_x Q$$

where Ω_x is obtained from

$$\Omega_x = -R \tan\theta \tag{5.3}$$

or

$$\Omega_x = \frac{2R\left[\lambda_1\lambda_3 - \lambda_0\lambda_2\right]}{\lambda_0^2 - \lambda_1^2 - \lambda_2^2 + \lambda_3^2} \tag{5.5}$$

Table 6. Time Development of Plane-Fixed Transformation Matrix Parameters

$$\dot{\lambda}_0 = -\tfrac{1}{2}\left\{ \lambda_2 Q + \frac{\lambda_3 R}{\left[\lambda_0^2 - \lambda_1^2 - \lambda_2^2 + \lambda_3^2\right]} \right\}$$

$$\dot{\lambda}_1 = -\tfrac{1}{2}\left\{ \lambda_3 Q + \frac{\lambda_2 R}{\left[\lambda_0^2 - \lambda_1^2 - \lambda_2^2 + \lambda_3^2\right]} \right\} \qquad (5.17)$$

$$\dot{\lambda}_2 = +\tfrac{1}{2}\left\{ \lambda_0 Q + \frac{\lambda_1 R}{\left[\lambda_0^2 - \lambda_1^2 - \lambda_2^2 + \lambda_3^2\right]} \right\}$$

$$\dot{\lambda}_3 = +\tfrac{1}{2}\left\{ \lambda_1 Q + \frac{\lambda_0 R}{\left[\lambda_0^2 - \lambda_1^2 - \lambda_2^2 + \lambda_3^2\right]} \right\}$$

or

$$\dot{\phi} = 0$$

$$\phi = 0$$

$$\dot{\psi} = \frac{R}{cos(\theta)} \qquad (3.16)$$

$$\dot{\theta} = Q \qquad (3.18)$$

(The above expressions (5.17) and (3.16) are singular near the vertical. Use body-fixed coordinates instead. See the discussion in the text.)

Table 7. Aeroballistic Equations

These are obtained by letting $\Omega_x = 0$, and assuming no products of inertia. A further simplification can be made by letting $I_{yy} = I_{zz} = I_t$ (axial symmetry).

$$\dot{U} = \frac{F_x}{m} + g_x^M - QW + RV$$

$$\dot{V} = \frac{F_y}{m} + g_y^M - RU \qquad (5.1),\ (5.8)$$

$$\dot{W} = \frac{F_z}{m} + g_z^M + QU$$

The components of gravitational acceleration are obtained using T^{-1}. For the flat earth approximation, use (5.9) and (5.10).

$$L = I_{xx}\dot{P} \qquad (5.15)$$

$$M = +I_t\dot{Q} + I_{xx}RP$$

$$N = +I_t\dot{R} - I_{xx}PQ$$

Table 8. Time Development of the Aeroballistic Transformation Matrix Parameters

$$\dot{\lambda}_0 = \tfrac{1}{2}\left[- Q\lambda_2 - R\lambda_3 \right]$$

$$\dot{\lambda}_1 = \tfrac{1}{2}\left[- Q\lambda_3 + R\lambda_2 \right]$$

$$\dot{\lambda}_2 = \tfrac{1}{2}\left[+ Q\lambda_0 - R\lambda_1 \right] \qquad (5.16\mathrm{B})$$

$$\dot{\lambda}_3 = \tfrac{1}{2}\left[+ Q\lambda_1 + R\lambda_0 \right]$$

or

$$\dot{\psi} = \frac{\left(Q\ sin(\phi) + R\ cos(\phi)\right)}{cos(\theta)} \qquad (3.16)$$

$$\dot{\phi} = \left(Q\ sin(\phi) + R\ cos(\phi)\right) tan(\theta) \qquad (3.17),(5.1)$$

$$= \dot{\psi}\sin(\theta)$$

$$\dot{\theta} = Q\ cos(\phi) - R\ sin(\phi) \qquad (3.18)$$

(The Euler angle expressions (3.16) and (3.17) have a singularity at $\pm\pi/2$.)

Equation (5.16B) comes from letting P=0 in (5.16).

6. INTEGRATION OF EQUATIONS OF MOTION

6.1 Plane-Fixed Equations

Recall for the plane-fixed case, the force equations from Table 5 are

$$
\begin{aligned}
\dot{U} &= \frac{F_x}{m} + g_x^M - QW + RV \\
\dot{V} &= \frac{F_y}{m} + g_y^M - RU + \Omega_x W \\
\dot{W} &= \frac{F_z}{m} + g_z^M - \Omega_x V + QU
\end{aligned}
\tag{6.1}
$$

where Ω_x is given by (5.13) and g^M is given by (5.9) for a flat earth or more generally by $g^M = T^{-1} g^I$, where the subscript I refers to an inertial frame. These expressions are readily integrated numerically.

For the plane-fixed case, the moment equations from Table 5 can be put into an uncoupled form for integration, where $I_t \equiv I_{yy} = I_{zz}$.

$$
\begin{aligned}
\dot{P} &= \frac{L}{I_{xx}} \\
\dot{Q} &= \frac{1}{I_t}\left[M - I_{xx} RP + I_t R\,\Omega_x\right] \\
\dot{R} &= \frac{1}{I_t}\left[N + I_{xx} QP - I_t Q\,\Omega_x\right]
\end{aligned}
\tag{6.2}
$$

These are readily integrated numerically.

6.2 Body-Fixed Equations

For body-fixed case, Ω_x = P and from (5.8)

$$\dot{U} = \frac{F_x}{m} + g_x^M - QW + RV$$

$$\dot{V} = \frac{F_y}{m} + g_y^M - RU + PW \tag{6.3}$$

$$\dot{W} = \frac{F_z}{m} + g_z^B - PV + QU$$

which are readily integrated numerically.

For the body-fixed case, equations (5.15) may be written

$$\dot{P} = \frac{1}{I_{xx}}\left[L + I_{yx}\dot{Q} + I_{xz}\dot{R} - f_1\left(P,Q,R\right)\right]$$

$$\dot{Q} = \frac{1}{I_{yy}}\left[M + I_{xy}\dot{P} + I_{yz}\dot{R} - f_2\left(P,Q,R\right)\right] \tag{6.4}$$

$$\dot{R} = \frac{1}{I_{zz}}\left[N + I_{xz}\dot{P} + I_{zy}\dot{Q} - f_3\left(P,Q,R\right)\right]$$

where

$$f_1 = \left[I_{zz} - I_{yy}\right]QR + I_{yz}\left[R^2 - Q^2\right] - I_{zx}QP + I_{xy}RP$$

$$f_2 = \left[I_{xx} - I_{zz}\right]RP + I_{yz}QP + I_{zx}\left[P^2 - R^2\right] - I_{xy}QR \tag{6.5}$$

$$f_3 = \left[I_{yy} - I_{xx}\right]PQ - I_{yz}PR + I_{zx}QR + I_{xy}\left[Q^2 - P^2\right]$$

As written, these equations are coupled. If the products of inertia vanish, the numerical integration is straight forward because the equations become uncoupled in the derivatives, as in (6.2) above. Even if they do not vanish, the products of inertia I_{xy}, I_{xz} and I_{yz} are generally quite small. This suggests a simple approximation. The equations could be solved by using the derivatives dP/dt, dQ/dt and dR/dt on the right side of equation (6.2) from the previous time step. Since the products of inertia are typically small, this approximation should be adequate in practice. For more precision, the results could be iterated. That is, the results for the derivatives dP/dt, dQ/dt and dR/dt on the left side could be put back into right side to obtain a better approximation before performing an integration.

If the products of inertia are not small or if one wishes to avoid this approximation, equations (6.4) and (6.5) can be put into the form

$$\begin{aligned}
L - f_1 &= +I_{xx}\dot{P} - I_{xy}\dot{Q} - I_{xz}\dot{R} \\
M - f_2 &= -I_{xy}\dot{P} + I_{yy}\dot{Q} - I_{yz}\dot{R} \\
N - f_3 &= -I_{xz}\dot{P} - I_{yz}\dot{Q} + I_{zz}\dot{R}
\end{aligned} \tag{6.6}$$

They can be solved simultaneously to uncouple the derivatives of P,Q and R by inverting the matrix of moment of inertia components. Formally we can write

$$\begin{vmatrix} \dot{P} \\ \dot{Q} \\ \dot{R} \end{vmatrix} = \begin{vmatrix} +I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & +I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & +I_{zz} \end{vmatrix}^{-1} \begin{vmatrix} L-f_1 \\ M-f_2 \\ N-f_3 \end{vmatrix} \tag{6.7}$$

Evaluating this inverse is somewhat tedious but the procedures for inverting a matrix are well known [1]. If we denote the matrix to be inverted by **J**

$$\mathbf{J}^{-1} = [\ \det \mathbf{J}\ ]^{-1} \begin{vmatrix} I_{yy}I_{zz} - I_{yz}^2 & I_{xz}I_{yz} + I_{xy}I_{zz} & I_{xy}I_{yz} + I_{xz}I_{yy} \\ I_{xz}I_{yz} + I_{xy}I_{zz} & I_{xx}I_{zz} - I_{xz}^2 & I_{xy}I_{xz} + I_{xx}I_{yz} \\ I_{xy}I_{yz} + I_{xz}I_{yy} & I_{xy}I_{xz} + I_{xx}I_{yz} & I_{xx}I_{yy} - I_{xy}^2 \end{vmatrix} \tag{6.8}$$

where

$$\det \mathbf{J} = I_{xx}I_{yy}I_{zz} - I_{xy}I_{yz}I_{xz} - I_{xz}I_{xy}I_{yz} - I_{yy}I_{xz}^2 - I_{zz}I_{xy}^2 - I_{xx}I_{yz}^2 \tag{6.9}$$

[1] Gelb, Arthur, *et al.*, Applied Optimal Estimation Theory", p 17, MIT Press, Cambridge, Mass, 1974.

6.3 Aeroballistic Equations

Recall for the aeroballistic case, the force equations from Table 7 are

$$\begin{aligned}
\dot{U} &= \frac{F_x}{m} + g_x^M - QW + RV \\
\dot{V} &= \frac{F_y}{m} + g_y^M - RU \\
\dot{W} &= \frac{F_z}{m} + g_z^M + QU
\end{aligned} \tag{6.10}$$

The moment equations from Table 7 are

$$\begin{aligned}
\dot{P} &= \frac{L}{I_{xx}} \\
\dot{Q} &= \frac{1}{I_t}\left[M - I_{xx} RP\right] \\
\dot{R} &= \frac{1}{I_t}\left[N + I_{xx} QP\right]
\end{aligned} \tag{6.11}$$

These are readily integrated numerically.

APPENDIX

ALGORITHMS FOR IMPLEMENTATION OF THE EQUATIONS OF MOTION IN SIX DEGREE OF FREEDOM COMPUTER SIMULATIONS

TRANSITIONS BETWEEN EULER ANGLES AND QUATERNIONS

The Initialization Problem

Since most people are more intuitively comfortable with Euler angles than with quaternions, the Euler angle to quaternion transformation can be used to input initial conditions in Euler angle format for the convenience of the user and convert to quaternions for internal use in a simulation if so desired. Conversely, quaternions used internally by a simulation can be converted to Euler angles prior to generating output, for the convenience of the user.

QUATERNIONS TO EULER ANGLES:

The Euler angles can be evaluated directly from the quaternions or indirectly from a rotation matrix that had been developed from the quaternions. Use (4.56).

Note that the denominator of the expressions for ψ and for θ in (4.56) can vanish. The algorithm in Table 2 in this document can handle this case correctly.

EULER ANGLES TO QUATERNIONS:

1) Evaluate the transformation matrix T from the Euler angles using (3.5). (With plane-fixed coordinates, the roll Euler angle ϕ must be set to zero. Alternatively, (3.22) can be used.)

2) Evaluate the quaternions using Table 1.

TRANSITIONS BETWEEN BODY-FIXED, PLANE-FIXED AND AEROBALLISTIC COORDINATES

Plane-fixed coordinates are more efficient for modeling spin-stabilized, unguided, rotationally symmetric projectiles. There is no component of gravity outside of the x-z plane in a flat earth model. Thus these coordinates are insensitive to gravity smearing because of roll. However, plane-fixed coordinates have a singularity for vertical trajectories and body-fixed coordinates are preferred for such trajectories. Body-fixed coordinates are also more appropriate for guided stages or other stages that don't have the required symmetry. Similarly, aeroballistic coordinates are also more efficient for axially-symmetric, spin-stabilized projectiles than body-fixed coordinates. Furthermore, the equations of motion are simpler for this choice than the other two candidate coordinate frames.

This document permits the development of a 6 degree of freedom simulation in which the coordinate frame can be changed from one stage to another to use the coordinate frame that is most appropriate or efficient in each particular stage of a trajectory simulation. Generally, other than changing the equations of motion, nothing special needs to be done when transitioning between one type of frame and another. There is an exception for transitioning to plane-fixed coordinates from other frames.

> *When transitioning to plane-fixed coordinates there is a discontinuous change in the orientation of the coordinate system, since ϕ must vanish in plane-fixed coordinates.* This requires the following adjustments:
>
> 1) The projectile angular velocity vector (P,Q,R) must be temporarily transformed to non-moving (i.e., inertial) coordinates using the last value of the rotation matrix.
>
> 2) A new rotation matrix T must be generated. (If the quaternion representation is being used, the equivalent Euler angles must be regenerated first, as shown in (4.56).) Set coordinate frame Euler roll angle ϕ to zero, retaining the regenerated pitch and yaw angles. Recalculate the rotation matrix with these new Euler angles using (3.5) or (3.22).
>
> 3) Use this matrix to rotate the projectile angular velocity vector in the non-moving frame back to the moving (plane-fixed) frame to obtain the new P, Q and R.
>
> 4) If using the quaternion formalism, regenerate the quaternions from the rotation matrix using Table 1.

Resume calculations using the appropriate equations of motion for the type of coordinate frame being used, for either the Euler angle or quaternion

representations described elsewhere in this document. The pitch θ and yaw ψ Euler angles for the body and for plane-fixed axes will be identical. The plane-fixed frame roll Euler angle ϕ is a constant and identically zero. The roll Euler angle for the aeroballistic frame is not constant but generally will not vary much from its value when transition occurred.

When using plane-fixed or aeroballistic coordinates, the body angular velocity component P must be reconstructed for use with the aerodynamics. When transitioning back to body-fixed coordinates, a roll angle can be restored by first constructing the inverse (i.e., transpose) of the roll rotation matrix from (3.3). The new full roation matrix is obtained by taking the matrix product of the existing plane-fixed rotation matrix and the roll matrix, in that order. This procedure will not corrupt the other variables in the equations of motion.

BODY-FIXED COORDINATES USING QUATERNIONS

Rotation matrix from body-fixed to inertial coordinates:

Calculate $T(\lambda_0, \lambda_1, \lambda_2, \lambda_3)$ from (4.40).

The inverse rotation matrix, from inertial coordinates to body-fixed, is obtained by taking the transpose of this matrix.

Time derivatives of the quaternions:

Calculate $\dot{\lambda}_0, \dot{\lambda}_1, \dot{\lambda}_2, \dot{\lambda}_3$ from (5.16).

Other Equations of Motion (See Tables 3 and 4.)

a) Force equations: Use (5.8) with $\Omega_x = P$.

See (5.1). Obtain components of gravity in body-fixed frame from the inertial frame by using T^{-1}. For flat earth, use (5.9).

b) Moment equations: Use (6.4) and (6.5) or (6.9) to (6.11).

PLANE-FIXED COORDINATES USING QUATERNIONS

Rotation matrix from plane-fixed to inertial coordinates:

Calculate $T(\lambda_0, \lambda_1, \lambda_2, \lambda_3)$ using (4.40)

The inverse rotation matrix, from inertial coordinates to body-fixed, is obtained by taking the transpose of this matrix.

Time derivatives of the quaternions:

Calculate $\dot{\lambda}_0, \dot{\lambda}_1, \dot{\lambda}_2, \dot{\lambda}_3$ from (5.17).

with

$$\Omega_x = \frac{2R\left[\lambda_1\lambda_3 - \lambda_0\lambda_2\right]}{\lambda_0^2 - \lambda_1^2 - \lambda_2^2 + \lambda_3^2}$$

from (5.5). There is a constraint in (4.57).

Other Equations of Motion (See Tables 5 and 6):

a) Force equations: Use (5.8) with Ω_x from (5.5). See above.

See (5.1). Obtain components of gravity in body-fixed frame by using T^{-1}. For flat earth, use (5.9). Because of the constraint (4.57), there is no y component of gravity.

b) Moment equations: Use (6.2).

AEROBALLISTIC COORDINATES USING QUATERNIONS

Rotation matrix from body-fixed to inertial coordinates:

Calculate $T(\lambda_0, \lambda_1, \lambda_2, \lambda_3)$ from (4.40).

The inverse rotation matrix, from inertial coordinates to body-fixed, is obtained by taking the transpose of this matrix.

Time derivatives of the quaternions:

Calculate $\dot{\lambda}_0, \dot{\lambda}_1, \dot{\lambda}_2, \dot{\lambda}_3$ from (5.16B), or (5.16) with P=0.

Other Equations of Motion (See Tables 7 and 8.)

a) Force equations: Use (5.8) with $\Omega_x = P$.

See (5.1). Obtain components of gravity in body-fixed frame from the inertial frame by using T^{-1}. For flat earth, use (5.9).

b) Moment equations: Use (6.11).

BODY-FIXED COORDINATES USING EULER ANGLES

Rotation matrix from plane-fixed to inertial coordinates:

Calculate $T(\phi, \theta, \psi)$ from (3.5).

The inverse rotation matrix, from inertial coordinates to body-fixed, is obtained by taking the transpose of this matrix.

Time derivatives of the Euler angles:

$$\Omega_x = P \qquad \Omega_y = Q \qquad \Omega_z = R \tag{5.1}$$

$$\dot{\psi} = \frac{\left(Q \sin(\phi) + R \cos(\phi)\right)}{\cos(\theta)} \tag{3.16}$$

$$\dot{\phi} = P + \left(Q \sin(\phi) + R \cos(\phi)\right) \tan(\theta) \tag{3.17}$$

$$\dot{\theta} = Q \cos(\phi) - R \sin(\phi) \tag{3.18}$$

Other Equations of Motion (See Tables 3 and 4.)

a) Force equations: Use (5.8) with $\Omega_x = P$.

See (5.1). Obtain components of gravity in body-fixed frame by using T^{-1}. For flat earth, use (5.10).

b) Moment equations: Use (6.4) and (6.5) or (6.7)-(6.9).

PLANE-FIXED COORDINATES USING EULER ANGLES

Rotation matrix from plane-fixed to inertial coordinates:

Calculate $\mathbf{T}(\phi=0, \theta, \psi)$ from (3.5).

The inverse rotation matrix, from inertial coordinates to body-fixed, is obtained by taking the transpose of this matrix. Note that ϕ is zero in (3.5).

Time derivatives of the Euler angles:

Use $\dot{\phi} = \phi = 0$ instead (3.17) and $\Omega_y = Q$ and $\Omega_z = R$.

$$\dot{\psi} = \frac{\mathbf{R}}{\cos(\theta)} \text{ from (3.16).}$$

$$\dot{\theta} = \mathbf{Q} \text{ from (3.18)}$$

Other Equations of Motion (See Tables 5 and 6.)

a) Force equations: Use (5.8) with Ω_x from (5.3).

See (5.1). Obtain components of gravity in plane-fixed frame by using $\mathbf{T}^{-1}$. For flat earth, use (5.10) with $\phi = 0$.

b) Moment equations: Use (6.2).

AEROBALLISTIC COORDINATES USING EULER ANGLES

Rotation matrix from plane-fixed to inertial coordinates:

Calculate $T(\phi, \theta, \psi)$ from (3.5).

The inverse rotation matrix, from inertial coordinates to body-fixed, is obtained by taking the transpose of this matrix.

Time derivatives of the Euler angles:

$$\Omega_x = 0 \qquad \Omega_y = Q \qquad \Omega_z = R \tag{5.1}$$

$$\dot{\psi} = \frac{\left(Q \sin(\phi) + R \cos(\phi)\right)}{cos(\theta)} \tag{3.16}$$

$$\dot{\phi} = \left(Q \sin(\phi) + R \cos(\phi)\right) \tan(\theta) \tag{3.17}$$

$$= \dot{\psi} \sin(\theta)$$

$$\dot{\theta} = Q\, cos(\phi) - R \sin(\phi) \tag{3.18}$$

Other Equations of Motion (See Tables 7 and 8.)

a) Force equations: Use (5.8) with $\Omega_x = 0$.

See (5.1). Obtain components of gravity in body-fixed frame by using T^{-1}. For flat earth, use (5.10).

b) Moment equations: Use (6.11).

CONSTRAINTS, LIMITATIONS AND PRACTICAL REQUIREMENTS

I. Normalization constraint on quaternions:

We require $\lambda_0^2 + \lambda_1^2 + \lambda_2^2 + \lambda_3^2 = 1$ from (4.29).

At each integration time step (or at least at frequent intervals), normalize by dividing each λ_i by

$$N = \left[\lambda_0^2 + \lambda_1^2 + \lambda_2^2 + \lambda_3^2\right]^{-1/2}$$

II. Constraint for plane-fixed coordinates:

We require $\lambda_2 \lambda_3 + \lambda_0 \lambda_1 = 0$ from (4.57).

Check this constraint regularly. If it begins to fail:

a) regenerate the Euler angles from the matrix T using (4.56),
b) set $\phi = 0$, and
c) regenerate quaternions from (4.42).

III. Euler angle singularity:

Terminate simulation if θ is too close to ± 90 degrees because of the singularity at that angle. See (3.16) and (3.17). (N.B., the Euler angle rotations of roll, pitch and yaw may be chosen to move the singularity to occur along the horizontal rather than the vertical axes. A better solution is to use quaternions with body-fixed coordinates. For plane-fixed coordinates, see paragraph IV below.

IV. Quaternion singularity for plane-fixed coordinates:

This singularity is similar to III except it only occurs for plane-fixed coordinates and not for body-fixed or aeroballistic coordinates. Body-fixed coordinates chould be used for vertical trajectories rather than plane-fixed.

V. Axial symmetry requirement for plane-fixed and aeroballistic coordinates:

a) $I_{yy} = I_{zz}$

b) Products of inertia I_{xy}, I_{xz}, I_{yz}, I_{yx}, I_{zx}, I_{zy} all vanish.

c) Aerodynamic coefficients do not depend on roll angle.

DISTRIBUTION LIST

Commander
Armaments Research, Development and Engineering Center
US Army Tank Automotive and Armaments Command
ATTN:

AMSTA-AR-AE,	B. Bushey
AMSTA-AR-AET,	W. Ebihara
AMSTA-AR-AET-A,	M. Amoruso (15)
	E. F. Brown
	S. Chung
	A. Farina
	A. Fiorellini
	T. Forte
	J. Grau
	S. Han
	H. Hudgins
	W. Koenig
	W. Kuhnle
	C. Livecchia
	G. Malejko
	J. Murnane
	C. Ng
	D. Pedersen
	J. Snyder
	W. Toledo
	R. Trohanowsky
	J. Thomasovich
	E. Vazquez
	J. Whyte
	B. Wong
	L. Yee
AMSTA-AR-FSP-A,	T. Haritos
	R. Collette
	K. Kendl
	M. Rosenbluth

Picatinny Arsenal, NJ 07806-5000